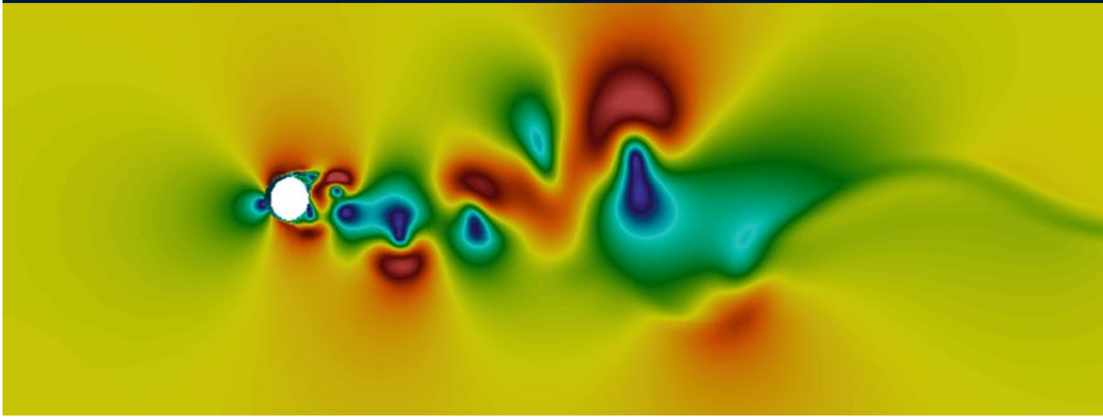
MS&A – Modeling, Simulation and Applications 17



Peter Benner · Mario Ohlberger
Anthony Patera · Gianluigi Rozza
Karsten Urban *Editors*

Model Reduction of Parametrized Systems

Springer

MS&A

Volume 17

More information about this series at http://www.springer.com/series/8377

Peter Benner • Mario Ohlberger • Anthony Patera •
Gianluigi Rozza • Karsten Urban
Editors

Model Reduction of Parametrized Systems

Springer

Editors

Peter Benner
Computational Methods in Systems
and Control Theory
Max Planck Institute for Dynamics
of Complex Technical Systems
Magdeburg, Germany

Mario Ohlberger
Institute for Computational and Applied
Mathematics
University of Münster
Münster, Germany

Anthony Patera
Massachusetts Institute of Technology
MIT
Cambridge
Massachusetts, USA

Gianluigi Rozza
SISSA mathLab
International School for Adv. Studies
Trieste
Italy

Karsten Urban
Institute of Numerical Mathematics
Ulm University
Ulm
Germany

ISSN 2037-5255 ISSN 2037-5263 (electronic)
MS&A – Modeling, Simulation & Applications
ISBN 978-3-319-58785-1 ISBN 978-3-319-58786-8 (eBook)
DOI 10.1007/978-3-319-58786-8

Library of Congress Control Number: 2017947450

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.
The publisher, the authors and the editors are safe to assume that the advice and information in this book are believed to be true and accurate at the date of publication. Neither the publisher nor the authors or the editors give a warranty, express or implied, with respect to the material contained herein or for any errors or omissions that may have been made. The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Cover illustration: velocity magnitude field around a cylinder immersed in an unsteady viscous flow at Re = 4000.
Courtesy of S. Hijazi, G. Stabile, A. Mola, G. Rozza, SISSA mathLab, 2017

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

This volume contains selected peer-reviewed contributions from the MoRePaS conference, Model Reduction of Parametrized Systems, third edition, held at SISSA, International School for Advanced Studies, Trieste, Italy on October 13–16, 2015: http://www.sissa.it/morepas2015, following its 2009, first edition, in Münster (hosted by the Westfälische Wilhelms-Universität Münster, Germany) and 2012, second edition, Schloss Reisensburg (hosted by Ulm University, Germany). The next MoRePaS 2018 will be in Nantes, France.

The MoRePaS workshop series aims to foster international exchange of new concepts and ideas in numerical analysis, applied mathematics, engineering, scientific computing, and programming with respect to the following topics:

- Reduced basis methods
- Proper orthogonal decomposition
- Proper generalized decomposition
- Approximation theory related to model reduction
- Learning theory and compressed sensing
- Stochastic and high-dimensional problems
- System-theoretic methods
- Nonlinear model reduction
- Reduction of coupled problems/multiphysics
- Optimization and optimal control
- State estimation and control
- Reduced order models and domain decomposition methods
- Krylov subspace and interpolatory methods
- Application to real, industrial and complex problems

The model reduction community is growing rapidly and during the past decade has achieved a high level of visibility in the computational science and engineering (CSE) global community, as is evident from the 30 contributions selected to appear in this special volume, which cover a broad range of modern topics as well as applications. A further collection of open access posters related to MoRePaS 2015 is available at www.scienceopen.com, doi:10.14293/S2199-1006.1.SOR-MATH.CLI8YJR.v1.

The MoRePaS organization also supports a website (www.morepas.org) collecting research software, preprints, organization of annual PhD student summer schools, open positions and several other activities.

This book represents the current state of the art in developments in applied mathematics, computational mathematics and engineering to deal with the increase in the complexity of modelled systems, to improve parametric computing, to deal with uncertainty quantification and to develop real-time computing. The need for a computational collaboration between full order classical discretization techniques and reduced order methods is highlighted.

We would like to thank the MoRePaS 2015 scientific committee for the editorial revision of this volume, as well as the COST EU-MORNET network (www.eu-mor.net)—European Union Cooperation in Science and Technology Model Reduction Network (TD1307)—for the support. We would like to say a big "thank you" to all the contributors, especially the invited speakers at the conference. Special thanks are also due to Angela Vanegas and Francesca Bonadei from Springer Italia for their editorial assistance.

Trieste, Italy — Gianluigi Rozza
Magdeburg, Germany — Peter Benner
Münster, Germany — Mario Ohlberger
Cambridge, MA, USA — Anthony T. Patera
Ulm, Germany — Karsten Urban
June 2017

MoRePaS 2015 Group Picture at SISSA, Trieste, Italy

Contents

About the Editors

Peter Benner is director of the Max Planck Institute for Dynamics of Complex Technical Systems and head of the department "Computational Methods in Systems and Control Theory." Moreover, he is a professor at the TU Chemnitz and adjunct professor at the Otto von Guericke University Magdeburg. He serves on the editorial board of several scientific journals, including *Advances in Computational Mathematics* and the *SIAM Journal on Matrix Analysis and Applications*.

Mario Ohlberger is a full professor of Applied mathematics and Managing director of Applied Mathematics at the University of Münster's Institute for Analysis and Numerics. He is an associate editor of five mathematical journals, including the *SIAM Journal on Scientific Computing*. He is a member of the Center for Nonlinear Science, the Center for Multiscale Theory and Computation, and the Cluster of Excellence "Cells in Motion."

Anthony T. Patera is the Ford Professor of Engineering and a professor of Mechanical Engineering at MIT and co-director of the MIT Center for Computational Engineering. His research interests include partial differential equations, computational methods, model order reduction, a posteriori error estimation, and data assimilation. Professor Patera holds SB and SM degrees in Mechanical Engineering from MIT and a PhD in Applied Mathematics, also from MIT. He served as co-editor in chief of the journal *Mathematical Modeling and Numerical Analysis* from 2003 to 2012.

Gianluigi Rozza has been an Associate Professor of Numerical Analysis and Scientific Computing at SISSA, International School for Advanced Studies since 2014. He holds a degree in Aerospace Engineering from Politecnico di Milano (2002) and a PhD in Applied Mathematics at Ecole Polytechnique Federale de Lausanne (2005). He was a post-doctoral research associate at the Massachusetts Institute of Technology (MIT) Center for Computational Engineering (2006–08), then a Researcher and Lecturer at EPFL (2008–2012). He is the author of over 100 scientific publications and recipient of the 2014 ECCOMAS young investigator Jacques Louis Lions Award in Computational Mathematics for researchers under the age of 40. Professor Rozza has been an associate editor of the SIAM/ASA Journal of

Uncertainty Quantification since 2013, of the SIAM Journal of Numerical Analysis since 2015, and of Computing and Visualization in Science since 2016.

Karsten Urban is a full professor of Numerical Mathematics and director of the Scientific Computing Centre at Ulm University. He is managing editor in chief of Advances in Computational Mathematics and associate editor of several mathematical journals, including the SIAM Journal on Scientific Computing. Further, he is currently directing several interdisciplinary research projects.

Chapter 1
Two Ways to Treat Time in Reduced Basis Methods

Silke Glas, Antonia Mayerhofer, and Karsten Urban

Abstract In this chapter, we compare two ways to treat the time within reduced basis methods (RBMs) for parabolic problems: Time-stepping and space-time variational based methods. We briefly recall both concepts and review well-posedness, error control and model reduction in both cases as well as the numerical realization. In particular, we highlight the conceptual differences of the two approaches.

We provide numerical investigations focussing on the performance of the RBM in both variants regarding approximation quality, efficiency and reliability of the error estimator. Pro's and Con's of both approaches are discussed.

1.1 Introduction

Parametrized partial differential equations (PPDEs) often occur in industrial or financial applications. If simulations are required for many different values of the involved parameters ("multi-query"), fine discretizations that are needed for a good approximation may resolve in high dimensional systems and thus in (for many applications too) long computation times. The reduced basis method (RBM) is by now a well-known model reduction technique, which allows one to efficiently reduce the numerical effort for many PPDEs by precomputing a reduced basis in an offline phase (using a detailed model, sometimes called "truth") and evaluating the reduced model (for new parameter values) highly efficient in the online phase.

Here, we focus on time-dependent problems of parabolic type in variational formulation and describe two different approaches. The maybe more standard one is based upon a time-stepping scheme in the offline phase. The reduced basis is then usually formed by the POD-Greedy method [3, 5], which results in a reduced time-stepping system for the offline phase. The second approach that we wish to discuss,

S. Glas • K. Urban (✉)
Ulm University, Institute for Numerical Mathematics, Helmholtzstr. 20, 89081 Ulm, Germany
e-mail: silke.glas@uni-ulm.de; karsten.urban@uni-ulm.de

A. Mayerhofer
Ulm University, Institute for Mathematical Finance, Helmholzstr. 18, 89081 Ulm, Germany
e-mail: antonia.mayerhofer@uni-ulm.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_1

is based upon the space-time variational formulation of the parabolic problem, in which the time is taken as an additional variable for the variational formulation. This results in a Petrov-Galerkin problem in $d + 1$ dimensions (if d denotes the spatial dimension). The reduced basis is then formed by a standard Greedy approach resulting in a reduced space-time Petrov-Galerkin method [13, 14].

The aim of this paper is to provide a comparison of these two methods in order to identify similarities and conceptual differences of the two approaches. It complements and completes a recent similar comparison in [2] for discrete instationary problems.

The remainder of this chapter is organized as follows. We start in Sect. 1.2 by reviewing both variational formulations of parabolic PDEs. A brief survey of the RBM is contained in Sect. 1.3. Section 1.4 is devoted to the description of the POD-Greedy/time-stepping method for the RBM, whereas Sect. 1.5 contains the space-time RBM. Our numerical experiments as well as the comparisons are presented in Sect. 1.6. We finish with some conclusions in Sect. 1.7.

1.2 Variational Formulations of Parabolic Problems

Let $V \hookrightarrow H$ be separable Hilbert spaces with continuous and dense embedding. The inner products and induced norms are denoted by $(\cdot, \cdot)_H$, $(\cdot, \cdot)_V$ and $\| \cdot \|_H$, $\| \cdot \|_V$, respectively. We identify H with its dual yielding a Gelfand triple $V \hookrightarrow H \hookrightarrow V'$.

Let $0 < T < \infty$ be the final time, $I := (0, T)$ the time interval and $\Omega \subset \mathbb{R}^d$ an open spatial domain. We consider a linear, bounded operator $A \in \mathcal{L}(V, V')$ induced by a bilinear form $a(\cdot, \cdot) : V \times V \to \mathbb{R}$ as $\langle A\phi, \psi \rangle_{V' \times V} = a(\phi, \psi)$.[1] We require the bilinear form to satisfy the following properties

$$|a(\phi, \psi)| \leq M_a \|\phi\|_V \|\psi\|_V, \quad \phi, \psi \in V \qquad \text{(boundedness)} \tag{1.1a}$$

$$a(\phi, \phi) + \lambda_a \|\phi\|_H^2 \geq \alpha_a \|\phi\|_V^2, \qquad \phi \in V \qquad \text{(Gårding inequality)} \tag{1.1b}$$

with constants $M_a < \infty$, $\alpha_a > 0$, $\lambda_a \geq 0$. Then, we consider the parabolic initial value problem of finding $u(t) \in V$, $t \in I$ a.e., such that

$$\dot{u}(t) + Au(t) = g(t), \ \text{ in } V', \quad u(0) = u_0 \text{ in } H, \tag{1.2}$$

where $g \in L_2(I; V')$ and $u_0 \in H$ are given.

[1] For simplicity, we restrict ourselves to Linear Time-Invariant (LTI) systems. However, much of what is said also holds for time-variant operators $A(t)$.

1.2.1 Semi-variational Formulation

The maybe more standard approach consists of multiplying (1.2) with a test function $\phi \in V$ and form the inner product in H (i.e., in space only). This leads to an evolution problem in V', i.e.,

$$\text{find } u(t) \in V: \ (\dot{u}(t), \phi)_H + a(u(t), \phi) = (g(t), \phi)_H, \quad \phi \in V, \ t \in I \text{ a.e.} \tag{1.3}$$

It is well-known that (1.3) is well-posed thanks to (1.1), see e.g. [15, Theorem 26.1]. Since the variational formulation is w.r.t. space only, we call it *semi-variational*.

1.2.2 Space-Time Variational Formulation

We now follow [12] for the description of a variational formulation of (1.2) w.r.t. space and time. This approach leads to a variational problem with different trial and test spaces $\mathbb{X}$ and $\mathbb{Y}$, both being Bochner spaces, namely

$$\mathbb{X} := L_2(I; V) \cap H^1(I; V') = \{v \in L_2(I; V): \ v, \dot{v} \in L_2(I; V')\}, \quad \mathbb{Y} := L_2(I; V) \times H,$$

with norms $\|w\|_{\mathbb{X}}^2 := \|w\|_{L_2(I;V)}^2 + \|\dot{w}\|_{L_2(I;V')}^2$, $w \in \mathbb{X}$, and $\|v\|_{\mathbb{Y}}^2 := \|v_1\|_{L_2(I;V)}^2 + \|v_2\|_H^2$, $v = (v_1, v_2) \in \mathbb{Y}$. Since $\mathbb{X} \hookrightarrow C(I; H)$, see, e.g. [15, Theorem 25.5], $u(0) \in H$ is well-defined for $u \in \mathbb{X}$.

The space-time variational formulation arises by multiplying (1.2) with test functions $v \in \mathbb{Y}$ and integrating w.r.t. time and space. This yields the linear operator $B \in \mathcal{L}(\mathbb{X}, \mathbb{Y}')$ defined as $\langle Bu, v\rangle_{\mathbb{Y}'\times\mathbb{Y}} = b(u, v)$ by (we omit the dependency on t in the integrands in the sequel)

$$b(u, v) := \int_I \langle \dot{u} + Au, v_1\rangle_{V'\times V} dt + (u(0), v_2)_H, \quad u \in \mathbb{X}, v = (v_1, v_2) \in \mathbb{Y},$$

i.e., $b(\cdot, \cdot) : \mathbb{X} \times \mathbb{Y} \to \mathbb{R}$ and the right-hand side $f \in \mathbb{Y}'$ is defined by

$$\langle f, v\rangle_{\mathbb{Y}'\times\mathbb{Y}} := \int_I \langle g, v_1\rangle_{V'\times V} dt + (u_0, v_2)_H, \quad v = (v_1, v_2) \in \mathbb{Y}.$$

Then, the space-time variational formulation of (1.2) reads

$$\text{find } u \in \mathbb{X}: \ \langle Bu, v\rangle_{\mathbb{Y}'\times\mathbb{Y}} = \langle f, v\rangle_{\mathbb{Y}'\times\mathbb{Y}}, \quad \forall\, v \in \mathbb{Y}. \tag{1.4}$$

Again, (1.1) yields well-posedness, e.g. [12, Theorem 5.1]. In fact, the operator B is boundedly invertible. The injectivity of the operator B is equivalent to

$$\beta_b := \inf_{w\in\mathbb{X}} \sup_{v\in\mathbb{Y}} \frac{|b(w,v)|}{\|w\|_{\mathbb{X}}\,\|v\|_{\mathbb{Y}}} > 0, \quad \text{(inf-sup condition)}. \tag{1.5}$$

1.3 Parametrized Problems and the RBM

We introduce a general notation here, which will then be specified for both variational formulations. Let $\mathcal{X}, \mathcal{Y}$ be Hilbert spaces, $\mu \in \mathcal{P} \subset \mathbb{R}^P$ a parameter and consider *parametric* forms $c : \mathcal{X} \times \mathcal{Y} \times \mathcal{P} \to \mathbb{R}$ as well as $h : \mathcal{Y} \times \mathcal{P} \to \mathbb{R}$. Then, the parameterized Petrov-Galerkin problem reads

$$\text{find } u(\mu) \in \mathcal{X} : \ c(u(\mu), v; \mu) = h(v;\mu) \quad \forall\ v \in \mathcal{Y}. \tag{1.6}$$

This framework obviously also includes the elliptic case, where $\mathcal{Y} = \mathcal{X}$. We briefly review the main ingredients of the RBM and refer to [6, 11] for recent surveys.

It is always assumed that a detailed discretization is available in the following sense: Let $\mathcal{X}^{\mathcal{N}} \subset \mathcal{X}$ and $\mathcal{Y}^{\mathcal{N}} \subset \mathcal{Y}$ be subspaces of finite, but large, dimension $\mathcal{N}$. The detailed problem (sometimes called "truth") then reads

$$\text{find } u^{\mathcal{N}}(\mu) \in \mathcal{X}^{\mathcal{N}} : \ c(u^{\mathcal{N}}(\mu), v^{\mathcal{N}}; \mu) = h(v^{\mathcal{N}};\mu) \quad \forall\ v^{\mathcal{N}} \in \mathcal{Y}^{\mathcal{N}}. \tag{1.7}$$

The "truth" solution $u^{\mathcal{N}}(\mu)$ is always assumed to be sufficiently close to $u(\mu)$. For well-posedness and stability of (1.7), a uniform inf-sup condition is required, e.g. [9].

The next step is the computation of a reduced basis formed by "snapshots" $\xi^i := u^{\mathcal{N}}(\mu^i)$, $1 \le i \le N \ll \mathcal{N}$, where the snapshot parameters $\mu^i \in \mathcal{P}$ are e.g. determined by a Greedy procedure w.r.t. an efficiently computable error estimate $\Delta_N(\mu)$. The reduced trial space is then defined as $\mathcal{X}_N := \operatorname{span}\{\xi^1, \dots, \xi^N\}$ and one needs some stable (possibly parameter-dependent) test space $\mathcal{Y}_N(\mu)$. The reduced problem reads

$$\text{find } u_N(\mu) \in \mathcal{X}_N : \ c(u_N(\mu), v_N; \mu) = h(v_N;\mu) \quad \forall\ v_N \in \mathcal{Y}_N(\mu). \tag{1.8}$$

In the above setting, it is easy to derive an error estimate which also results in the required error estimator $\Delta_N(\mu)$ defined by

$$\|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_{\mathcal{X}} \le \frac{1}{\beta_c}\|r_N(\cdot;\mu)\|_{\mathcal{Y}'} =: \Delta_N(\mu), \tag{1.9}$$

where $r_N(v;\mu) := h(v;\mu) - c(u_N(\mu), v;\mu) = c(u^{\mathcal{N}}(\mu) - u_N(\mu), v;\mu)$, $v \in \mathcal{Y}^{\mathcal{N}}$, is the *residual* and β_c is the inf-sup constant associated with the bilinear form c.

We call (1.8) *online efficient* if it can be solved with complexity independent of $\mathcal{N}$. In order to reach that goal, the following assumption is crucial: The forms are assumed to be separable in the parameter (sometimes called *affine decomposition*),

$$c(u, v; \mu) = \sum_{q=1}^{Q_c} \theta_q^c(\mu)\, c_q(u, v), \qquad h(v; \mu) = \sum_{q=1}^{Q_h} \theta_q^h(\mu)\, h_q(v) \tag{1.10}$$

for some $Q_c, Q_h \in \mathbb{N}$, functions $\theta_q^c, \theta_q^h : \mathcal{P} \to \mathbb{R}$ and parameter-independent forms c_q, h_q that can be precomputed in the offline phase. The parameter-dependent functions θ_q^c, θ_q^h are assumed to be computable online efficient, i.e., with complexity independent of $\mathcal{N}$.

1.4 Reduced Basis Methods with POD-Greedy

We start from the semi-variational formulation (1.3) and apply a semi-discretization in time, known as Rothe's method. To this end, set $\Delta t := \frac{T}{K}$ for some $K > 1$, $t^k := k\,\Delta t$ and we seek some approximation $u^k \approx u(t^k)$, where we omit the μ-dependency to shorten notation. This leads to a sequence of elliptic (time-independent) ordinary differential equations starting with $u^0 := u_0$. The standard θ-scheme then reads

$$\begin{aligned}\frac{1}{\Delta t}\big(u^{k+1} - u^k, v\big)_H &+ a(\theta u^{k+1} + (1-\theta)u^k, v; \mu) \\ &= \theta g(v, t^{k+1}; \mu) + (1-\theta) g(v, t^k; \mu), \quad v \in V.\end{aligned}$$

"Truth" The next step is a discretization in space by a standard Galerkin method using finite-dimensional spaces $V_h \subset V$ with large $\dim(V_h) = \mathcal{N}_h \in \mathbb{N}$. Then, the detailed or "truth" problem for a given parameter $\mu \in \mathcal{P}$ reads for an initial value $u_h^0 := \mathrm{Proj}_{V_h} u^0$ to find $u_h^{k+1}(\mu) \in V_h$, such that for $v_h \in V_h$,

$$\begin{aligned}&(u_h^{k+1}, v_h)_H + \Delta t \theta\, a(u_h^{k+1}, v_h; \mu) \\ &= (u_h^k, v_h)_H + \Delta t(1-\theta)\, a(u_h^k, v_h; \mu) + \theta g(v_h, t^{k+1}; \mu) + (1-\theta) g(v_h, t^k; \mu),\end{aligned}$$

for $0 \le k \le K-1$. If $V_h = \mathrm{span}\{\phi_i : i = 1, \ldots, \mathcal{N}_h\}$, the latter equation can be written in matrix-vector form as follows. Let $\underline{M}_h^{\text{space}} := [(\phi_i, \phi_j)_H]_{i,j=1,\ldots,\mathcal{N}_h}$ denote the spatial mass matrix and $\underline{A}_h^{\text{space}} := [a(\phi_i, \phi_j)]_{i,j=1,\ldots,\mathcal{N}_h}$ the stiffness matrix (we denote matrices and vectors by underlined symbols), then we look for

$$u_h^{k+1} = \sum_{i=1}^{\mathcal{N}_h} \alpha_i^{k+1}\, \phi_i, \qquad \underline{\alpha}^{k+1} := (\alpha_i^{k+1})_{i=1,\ldots,\mathcal{N}_h},$$

such that ($\underline{g}^k$ and $\underline{\alpha}_0$ being the expansion coefficients of $g(t^k)$ and u_0, respectively)

$$\begin{aligned}&(\underline{M}_h^{\text{space}} + \theta \Delta t \underline{A}_h^{\text{space}}(\mu))\underline{\alpha}^{k+1} \\ &= (\underline{M}_h^{\text{space}} + (1-\theta)\Delta t \underline{A}_h^{\text{space}})\underline{\alpha}^k + \Delta t(\theta \underline{g}^{k+1} + (1-\theta)\underline{g}^k), \end{aligned} \tag{1.11}$$

for $0 \leq k \leq K-1$ as well as $\underline{\alpha}^0 := \underline{\alpha}_0$. It is well-known that the θ-scheme is unconditionally stable for $\frac{1}{2} \leq \theta \leq 1$, whereas for $0 \leq \theta < \frac{1}{2}$ the space discretization has to satisfy additional properties, see e.g. [10, Theorem 11.3.1]. The choice $\theta = \frac{1}{2}$ results in the Crank-Nicolson scheme. Note, that (1.11) requires to solve a well-posed elliptic problem for each time step k, which easily follows from the assumption (1.1) on the bilinear form a and coercivity of $m(\phi, \psi) := (\phi, \psi)_H$. In fact, this implies that the matrix $\underline{M}_h^{\text{space}} + \theta \Delta t \underline{A}_h^{\text{space}}(\mu)$ is positive definite, e.g. [10, §11.3].

The system (1.11) is offline/online decomposable which is easily seen as long as the forms a and g are separable in the parameter. In fact, the mass inner product m is independent of the parameter.

Reduced Basis via POD-Greedy The reduced basis is computed by the POD-Greedy method shown in Algorithm 1. This is a combination of the standard Greedy algorithm for the parameter search and a Proper Orthogonal Decomposition (POD) in time to select the time step containing the maximal information of the trajectory for the given parameter.

Online Phase The POD-Greedy method produces a reduced space $V_N \subset V$ of possibly small dimension $N := N_{\text{POD-G}} \ll \mathcal{N}_h$. Then, a reduced basis approximation for a new parameter $\mu \in \mathcal{P}$ is determined by a corresponding time-stepping scheme as follows. The reduced initial value $u_N^0 \in V_N$ is computed by $(u_N^0 - u_0, v_N)_V = 0$ for all $v_N \in V_N$. Then, for $0 \leq k \leq K-1$, determine $u_N^{k+1}(\mu) \in V_N$ by

$$\frac{1}{\Delta t}(u_N^{k+1} - u_N^k, v_N)_H + a(\theta u_N^{k+1} + (1-\theta)u_N^k, v_N; \mu) = f(v_N; \mu), \quad v_N \in V_N.$$

Obviously, this amounts solving a sequence of K reduced problems online.

Algorithm 1 POD-Greedy algorithm [5]

```
Require: Given $N_{\max} > 0$, $\mathcal{P}_{\text{train}} \subset \mathcal{P}$, $\epsilon_{\text{tol}}$, $\ell = 1$
choose arbitrarily $\mu^\ell \in \mathcal{P}_{\text{train}}$; set $\Psi_\ell := \left\{ \frac{u^0(\mu^\ell)}{\|u^0(\mu^\ell)\|_V} \right\}$, $V_\ell := \text{span}(\Psi_\ell)$
while $\max_{\mu \in \mathcal{P}_{\text{train}}} \Delta_\ell(\mu) > \epsilon_{\text{tol}}$ do
    define $\mu^{\ell+1} := \text{argmax}_{\mu \in \mathcal{P}_{\text{train}}} \Delta_\ell(\mu)$
    define $\tilde{\psi}_{\ell+1} := \text{POD}\{u^k(\mu^{\ell+1}) - \text{Proj}_{V_\ell}(u^k(\mu^{\ell+1}))\}_{k=0,\dots,K}$
    define $\Psi_{\ell+1} := \text{orthonormalize}(\Psi_\ell \cup \{\tilde{\psi}_{\ell+1}\})$, $V_{\ell+1} := \text{span}(\Psi_{\ell+1})$, $\ell = \ell + 1$
end while
define $V_N := V_\ell$, $N_V := \dim(V_N)$
return $V_N, N_V$;
```

Error Estimator/Indicator As in the standard case in Sect. 1.3, an online efficient error estimator is needed both in Algorithm 1 and online for the desired certification of the RB approximation. Of course, such an estimator here also needs to incorporate the temporal evolution. In fact, there are several known choices for $\varDelta_N(\mu)$ in Algorithm 1. A standard estimator (bound) for the error $e^k(\mu) := u_h^k(\mu) - u_N^k(\mu)$ at final time $T = t^K$ is given by [4, Proposition 3.9]

$$\|e^K(\mu)\|_V \leq \|e^0(\mu)\|_V \left(\frac{\gamma_{\text{UB}}}{\alpha_{\text{LB}}}\right)^K + \sum_{k=0}^{K-1} \frac{\Delta t}{\alpha_{\text{LB}}} \left(\frac{\gamma_{\text{UB}}}{\alpha_{\text{LB}}}\right)^{K-k-1} \|r_N^k(\mu)\|_{V'} =: \varDelta_N^K(\mu). \tag{1.12}$$

Here, α_{LB} is a lower bound for the coercivity constant of the implicit part of the operator, γ_{UB} an upper bound for the continuity constant of the explicit part and $r_N^k(\mu)$ is the residual at time step t^k. It can easily be seen that $\varDelta_N^K$ is offline/online decomposable. There are some remarks in order.

Remark 1

(i) In our numerical experiments in Sect. 1.6 below, we use a finite element (FE) discretization. In that case, the estimator (1.12) can not be used. In fact, $\alpha_{\text{LB}} \ll \gamma_{\text{UB}}$, in our case $\varDelta_N^K(\mu) \approx 10^{119}$. This shows that $\varDelta_N^K(\mu)$ grows extremely quickly with increasing K for FE discretizations in $V = H_0^1(\varOmega)$, which makes (1.12) practically useless. Note, however, that for FV discretizations, one has $\alpha_{\text{LB}} = \gamma_{\text{UB}} = 1$, so that the estimator works often quite well.
(ii) For *symmetric* differential operators, the above estimate can be sharpened [4, Proposition 3.11]. It allows to extend (1.12) to a μ-dependent norm on the whole trajectory. ◇

According to this remark, we cannot use $\varDelta_N^K(\mu)$ here. Thus, we follow the analysis in [4] and consider a weighted (sometimes called "space-time") norm for $\underline{\omega} := (\omega_k)_{k=0,\dots,K-1}$, $\omega_k > 0$ and $\sum_{k=0}^{K-1} \omega_k = T$ defined as

$$|\boldsymbol{e}|_{\underline{\omega}}^2 := \sum_{k=0}^{K-1} \omega_k \|e^k\|_V^2, \qquad \boldsymbol{e} = (e^k)_{k=0,\dots,K-1}. \tag{1.13}$$

A corresponding error *indicator* is defined as

$$\varDelta_{N,\underline{\omega}}^{\text{PST}} := \left(\sum_{k=0}^{K-1} \omega_k \|r_N^k(\mu)\|_{V'}^2\right)^{1/2}. \tag{1.14}$$

The term "indicator" means that the error (in whatever norm) cannot be proven to be bounded in terms of $\varDelta_{N,\boldsymbol{\omega}}^{\text{PST}}$ (it is *not* known to be a bound). However, even though the error of the POD-Greedy scheme cannot be guaranteed to decay monotonically, exponential convergence can be shown under additional assumptions [3].

1.5 Space-Time Reduced Basis Methods

As we have seen in Sect. 1.2, the space-time variational formulation leads to a parameterized Petrov-Galerkin problem. Thus, the form is exactly as in the general RB-framework in (1.6). Note, that both $\mathbb{X} = H^1(I) \otimes V$ and $\mathbb{Y} = (L_2(I) \otimes V) \times H$ are tensor products, so that it is convenient to use the same structure for the detailed discretization, i.e., $\mathbb{X}^{\mathcal{N}} = S_{\Delta t} \otimes V_h$ and $\mathbb{Y}^{\mathcal{N}} = (Q_{\Delta t} \otimes V_h) \times V_h$,[2] where $V_h \subset V$ is the space discretization as in Sect. 1.4 above and $S_{\Delta t} \subset H^1(I)$ as well as $Q_{\Delta t} \subset L_2(I)$ are temporal discretizations of step size Δt [14].

Let us denote again by $V_h = \operatorname{span}\{\phi_1, \phi_2, \dots, \phi_{\mathcal{N}_h}\}$ the detailed space discretization (e.g. by piecewise linear finite elements for $V = H_0^1(\Omega)$). Moreover, let $S_{\Delta t} = \operatorname{span}\{\sigma^0, \sigma^1, \dots, \sigma^K\}$ and $Q_{\Delta t} = \operatorname{span}\{\tau^1, \tau^2, \dots, \tau^K\}$ be the bases in time (e.g. piecewise linear σ^i and piecewise constant τ^i on the same temporal mesh, with the additional σ^0 for the initial value at $t = 0$).

The dimension of the arising test and trial spaces coincide, i.e., $\dim(\mathbb{X}^{\mathcal{N}}) = (K+1)\mathcal{N}_h = \dim(\mathbb{Y}^{\mathcal{N}}) =: \mathcal{N}$. Exploiting the structure of the discretized spaces for the detailed solution $u^{\mathcal{N}} = \sum_{i=1}^{\mathcal{N}_h} \sum_{k=0}^{K} u_i^k \sigma^k \otimes \phi_i$ yields

$$\begin{aligned} b(u^{\mathcal{N}}, (\tau^k \otimes \phi_j, 0); \mu) &= \sum_{i=1}^{\mathcal{N}_h} [(u_i^k - u_i^{k-1})(\phi_i, \phi_j)_H + \frac{\Delta t}{2}(u_i^k + u_i^{k-1}) a(\phi_i, \phi_j; \mu)] \\ &= \underline{M}_h^{\text{space}}(u^k - u^{k-1}) + \Delta t \underline{A}_h^{\text{space}}(\mu) u^{k-1/2}, \end{aligned}$$

with mass and stiffness matrices $\underline{M}_h^{\text{space}}$, $\underline{A}_h^{\text{space}}(\mu)$ as above. On the right-hand side, we use a trapezoidal approximation as in [14] on a time grid $0 = t^0 < \cdots < t^K = T$, $I^\ell := [t^{\ell-1}, t^\ell) = \operatorname{supp}\{\tau^\ell\}$:

$$\begin{aligned} f((\tau^\ell \otimes \phi_j, 0); \mu) &= \int_I \langle g(t; \mu), \tau^\ell \otimes \phi_j(t, .)\rangle_{V' \times V} dt = \int_{I^\ell} \langle g(t; \mu), \tau^\ell(t) \phi_j \rangle_{V' \times V} dt \\ &\approx \frac{\Delta t}{2} \langle g(t^{\ell-1}; \mu) + g(t^\ell; \mu), \phi_j \rangle_{V' \times V}. \end{aligned}$$

It turns out that this particular choice for the discretization results (again) in the Crank-Nicolson scheme involving an additional projection of the initial value, which requires a CFL condition to ensure stability. A detailed investigation of stable space-time discretizations can be found in [1].

[2] It can be seen that $V_h \subset V \hookrightarrow H$ is in fact sufficient [8].

Since the space-time variational approach yields a standard Petrov-Galerkin problem, the reduced basis trial and test spaces $\mathbb{X}_N = \operatorname{span}\{\xi^1, \dots, \xi^N\}$, $\mathbb{Y}_N(\mu) := \operatorname{span}\{\eta^1(\mu), \dots, \eta^N(\mu)\}$ can be constructed exactly following the road map in Sect. 1.3. Hence, we end up with a reduced problem of the form (1.8). In matrix-vector form, the resulting system matrix $\underline{B}_N(\mu) := [b(\xi^i, \eta^j(\mu); \mu)]_{i,j=1,\dots,N}$ is of small dimension, but not symmetric. Moreover, $\underline{B}_N(\mu)$ is uniformly invertible provided that the inf-sup condition in (1.5) holds for the RB spaces. It is not difficult to show that the arising non-symmetric linear system can also be written as minimization problem or in terms of normal equations (see [8] for details and further applications).

If normal equations are used, no (parameter dependent) reduced test space computation is required: Let $\mathbb{X}^{\mathcal{N}} = \operatorname{span}\{\varphi_\ell : \ell = 1, \dots, \mathcal{N}\}$ and $\mathbb{Y}^{\mathcal{N}} = \operatorname{span}\{\psi_m : m = 1, \dots, \mathcal{N}\}$ be the detailed bases, denote by $\underline{Y}^{\mathcal{N}} := [(\psi_m, \psi_{m'})_{\mathbb{Y}}]_{m,m'=1,\dots,\mathcal{N}}$ the mass matrix of the test space $\mathbb{Y}^{\mathcal{N}}$ as well as the detailed system matrix by $\underline{B}^{\mathcal{N}}(\mu) := [b(\varphi_\ell, \psi_m; \mu)]_{\ell,m=1,\dots,\mathcal{N}}$. Next, denote by $\xi^j = \sum_{\ell=1}^{\mathcal{N}} c_\ell^j \varphi_\ell$, $\underline{C} := (c_\ell^j)_{\ell=1,\dots,\mathcal{N}, j=1,\dots,N} \in \mathbb{R}^{\mathcal{N} \times N}$ the expansion of the RB functions in terms of the detailed basis. Then,

$$\underline{B}_N(\mu) := \underline{C}^T \underline{B}^{\mathcal{N}}(\mu)(\underline{Y}^{\mathcal{N}})^{-1}(\underline{B}^{\mathcal{N}}(\mu))^T \underline{C}, \quad \underline{f}_N(\mu) := \underline{C}^T \underline{B}^{\mathcal{N}}(\mu)(\underline{Y}^{\mathcal{N}})^{-1}\underline{f}^{\mathcal{N}}(\mu),$$

where $\underline{f}^{\mathcal{N}}(\mu)$ contains the detailed basis coefficients of the right-hand side. The RB approximation $u_N(\mu) = \sum_{i=1}^N \alpha_i(\mu)\xi^i$, $\underline{\alpha}(\mu) := (\alpha_i(\mu))_{i=1,\dots,N}$, is then determined by the solution of the linear system of size N, i.e.

$$\underline{B}_N(\mu)\, \underline{\alpha}(\mu) = \underline{f}_N(\mu). \tag{1.15}$$

One can show that (1.15) admits an online/offline-separation, which is inherited from the separation of the forms a and g (in particular, we have $Q_b = Q_a$ and $Q_f = Q_g$). This means that (1.15) can be solved online efficient. Finally, the inf-sup stability of (1.15) is inherited from the detailed discretization.

1.6 Numerical Results

We provide some of our numerical investigations concerning the two approaches described above for a standard diffusion-convection-reaction problem with time dependent right-hand side. Since our focus is on the treatment of the time variable, we restrict ourselves to a 1d problem in space.

1.6.1 Data

Model Problem Let $d = 1$, $\Omega = (0,1)$ and $V := H_0^1(\Omega) \hookrightarrow L_2(\Omega) =: H$. Consider the time interval $I = (0, 0.3)$ and the parameter set $\mathcal{P} := [0.5, 1.5] \times [0,1] \times [0,1] \subset \mathbb{R}^3$. For a parameter $\mu = (\mu_1, \mu_2, \mu_3)^T \in \mathcal{P}$ find $u \equiv u(\mu)$ that solves

$$\dot{u} - \mu_1 u'' + \mu_2 u' + \mu_3 u = g \qquad \text{on } I \times \Omega, \tag{1.16a}$$

$$u(t,x) = 0 \qquad \forall\ (t,x) \in I \times \partial\Omega, \tag{1.16b}$$

$$u(0,x) = u_0(x) := \sin(2\pi x) \qquad \forall\ x \in \Omega. \tag{1.16c}$$

Set $g(t,x) := \sin(2\pi x)((4\pi^2 + 0.5)\cos(4\pi t) - 4\pi \sin(4\pi t)) + \pi \cos(2\pi x)\cos(4\pi t)$, which corresponds to the solution $u(t,x) := \sin(2\pi x)\cos(4\pi t)$ of (1.16) for the reference parameter $\mu^{\text{ref}} = (1, 0.5, 0.5) \in \mathcal{P}$. The parameter-separability is easily seen. We divide both Ω and I into 2^6 subintervals, i.e., $\mathcal{N}_h = 2^6 - 1$ and $K = 2^6$, but we also consider various values for K. The training set $\mathcal{P}_{\text{train}}$ is chosen as 17 equidistantly distributed points in $\mathcal{P}$ in each direction.

"True" Norms For the space-time RBM, the "true" error is measured in the natural discrete space-time norm [13]

$$|||v|||_{\mathcal{N}}^2 := \|\bar{v}\|_{L_2(I;V)}^2 + \|\dot{v}\|_{L_2(I;V')}^2 + \|v(T)\|_H^2, \qquad v \in \mathbb{X}^{\mathcal{N}} \subset \mathbb{X},$$

where $\bar{v} := \sum_{k=1}^{K} \tau^k \otimes \bar{v}^k \in L_2(I;V)$ and $\bar{v}^k := (\Delta t)^{-1} \int_{I^k} v(t)\, dt \in V$. Note, that $\|\bar{v}\|_{L_2(I;V)}$ is an $\mathcal{O}(\Delta t)$-approximation of $\|v\|_{L_2(I;V)}$ (due to the piecewise constant approximation $\bar{v}^k$).

For the POD-Greedy strategy, we consider the final time contribution $\|v(T)\|_V$ [corresponding to the left-hand side of (1.12)] as well as the "space-time norm" introduced in (1.13) for the specific weights $\omega_k := \Delta t$, $k = 0, \ldots, K-1$, plus the final time contribution as "true" error, i.e.

$$|v|_{\Delta t}^2 := |(v(t^k))_{k=0,\ldots,K-1}|_{\underline{\omega}}^2 + \Delta t\, \|v(T)\|_V^2 = \sum_{k=0}^{K} \Delta t \|v(t^k)\|_V^2.$$

This means that $|v|_{\Delta t}$ is an $\mathcal{O}(\Delta t^2)$-approximation of $\|v\|_{L_2(I;V)}$.

Remark 2 For later reference, we point out that $|v|_{\Delta t}$ is an $\mathcal{O}(\Delta t^2)$-approximation of $\|v\|_{L_2(I;V)}$, whereas $\|\bar{v}\|_{L_2(I;V)}$ is only of the order $\mathcal{O}(\Delta t)$. This means that it may happen that $|w|_{\Delta t} > |||w|||_{\mathcal{N}}$ even though $\|w\|_{\mathbb{X}} > \|w\|_{L_2(I;V)}$ for all $0 \neq w \in \mathbb{X}$. $\diamond$

Error Estimators/Greedy For the error estimation in the space-time RBM, we use the residual-based error estimator $\Delta_N(\mu)$ in (1.9) with numerically estimated lower bound for the inf-sup constant, $\beta_{\text{LB}} = 0.2$. Of course, this is a relatively rough bound (independent of μ!) and performing e.g. a Successive Constraint Method (SCM [7]) would improve the subsequent results. For the POD-Greedy scheme, we use the space-time error indicator $\Delta_N^{\text{PST}}(\mu)$ that arises from (1.14) by the choice $\omega_k \equiv \Delta t$ for the weights.

We emphasize that $\Delta_N(\mu)$ bounds the norm in $\mathbb{X} = H^1(I) \otimes V \subsetneq L_2(I; V)$, whereas $\Delta_N^{\text{PST}}(\mu)$ is "only" an indicator, in particular it is not known to be an upper bound for any norm. In view of (1.12), we could expect that $\|e^K(\mu)\|_V$ might be a candidate, or—since $|\cdot|_{\Delta t}$ is a discrete $L_2(I; V)$-norm—we could consider $\|e\|_{L_2(I;V)}$.

Comparison We conclude that a direct and fair comparison is not easy because of the described methodological differences. Table 1.1 collects these differences and our choice for the experiments.

1.6.2 Results

Greedy Training Within the framework of Table 1.1, the offline Greedy error decay of both variants is shown in Fig. 1.1. The red lines correspond to the space-time form, whereas the blue lines are for the POD-Greedy. Straight lines indicate the error, dotted ones the error estimator/indicator. The left figure shows the weak Greedy using the error estimators/indicators. We observe exponential decay w.r.t. the RB dimension N in both cases—as predicted by the theory. At a first glance, it seems that the decay of POD-Greedy is much faster than the one for space-time, i.e., the stopping criterium in the Greedy algorithm for tol $= 10^{-3}$ is reached at $N_{\text{POD-G}} = 7$ and $N_{\text{ST}} = 16$. Note, however, that the online effort is related to N in a different way for both variants, see below.

As mentioned above, the two variants are related to different norms. This is the reason why we performed a strong Greedy using the $|||\cdot|||_{\mathcal{N}}$-norm for both variants (right graph in Fig. 1.1). We see a similar behavior—again referring to the different online work load. It is interesting, though, that at least in these experiments, POD-Greedy works very well even for the full norm—without theoretically foundation though. However, we can also see from the results that $\Delta_N^{\text{PST}}(\mu)$ is not an error bound since it is below the "exact" error for some N. Recall that $\Delta_N^{\text{PST}}(\mu)$ arises

Table 1.1 Differences of space-time RBM and POD-Greedy sampling

	Snapshot space	RB space	"True" error	Error estimated by
Space-time	$\mathbb{X}^{\mathcal{N}}$	$\mathbb{X}_N$	$\|\|\|\cdot\|\|\|_{\mathcal{N}}$	Residual based, $\Delta_N(\mu)$ in (1.9)
POD-Greedy	$\{t^0, \dots, t^K\} \times V_h$	V_N	$\|\cdot\|_{\Delta t}$	Indicator $\Delta_N^{\text{PST}}(\mu)$ in (1.14) for $\omega_k \equiv \Delta t$

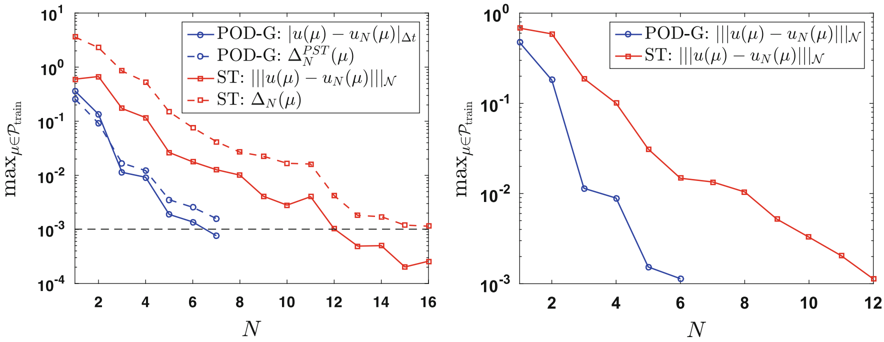


Fig. 1.1 RB-Greedy approximation error. *Red*: Space-Time (ST); *blue*: POD-Greedy (POD-G). *Left*: weak Greedy, *right*: strong Greedy w.r.t. $|||\cdot|||_{\mathcal{N}}$. *Lines* are plotted until the stopping criteria for tol $= 10^{-3}$ are reached in the while-loop (i.e., the error estimators in the next step are below tol)

from the upper bound $\Delta_N^K(\mu)$ in (1.12) by setting involved coercivity and continuity constants to 1. Moreover, note, that in the same spirit, we could easily lower the gap between the two red space-time lines by sharpen β_{LB}. The prescribed Greedy tolerance of 10^{-3} is reached for $N_{\text{POD-G}} = 6$ for POD-Greedy and for $N_{ST} = 12$ for space-time.

Online Phase We test the RB approximations for two cases, namely for $\mathcal{P}_{\text{test}}^{\text{sym}} := \{(\mu_1, 0, 0) : \mu_1 \in [0.5, 1.5]\}$ (symmetric case) and $\mathcal{P}_{\text{test}}^{\text{non}} := \{(0.5, \mu_2, 0.75) : \mu_2 \in [0, 1]\}$ (non-symmetric case). The results are shown in Fig. 1.2, the symmetric case in the top line, the non-symmetric in the bottom one.

First of all, both variants work as they should in the sense that the respective error measures are below the Greedy tolerance of 10^{-3}.

The "true" error (solid red lines) is slightly smaller for the space-time variant in the symmetric case and very close to each other in the non-symmetric case.

We also compare the POD-Greedy error measure $|\cdot|_{\Delta t}$. It is remarkable that this quantity is almost identical to $|||\cdot|||_{\mathcal{N}}$ for space-time. This means that the temporal derivative and the final time components of the solution are very-well approximated. Regarding the result that for POD-Greedy the $|\cdot|_{\Delta t}$-lines turn out to be above the $|||\cdot|||_{\mathcal{N}}$ one, we recall Remark 2.

Finally, the error estimators/indicators are plotted as dash-dotted black lines. We observe, that the effectivities[3] are of almost the same size. However, if we rely on the respective theory (i.e. $|||e|||_{\mathcal{N}}$ for space-time and $\|e(T)\|_V$ for POD-Greedy), the effectivity of space-time improves, see Fig. 1.3.

[3] It is somehow misleading to use the term "effectivity" within the POD-Greedy framework, since usually effectivity is the ratio of *error bound* and error.

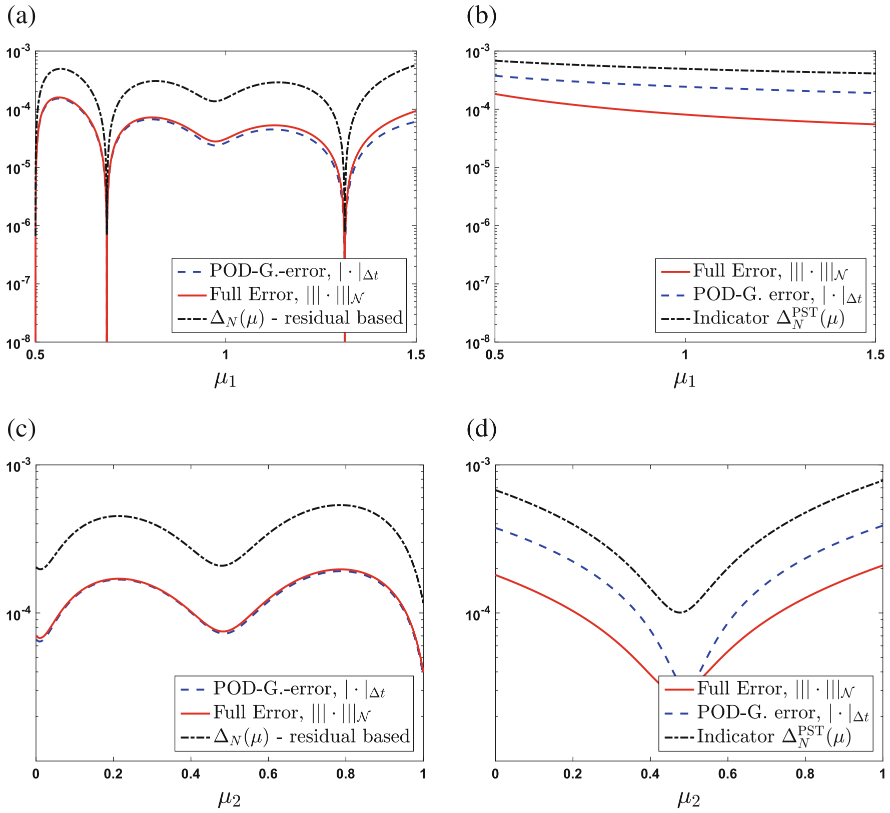


Fig. 1.2 RB approximation error on $\mathcal{P}_{\text{test}}$: Full error (*red*, *solid*), error estimator/indicator (*black*, *dash-dotted*) and the POD-Greedy error measure $|\cdot|_{\Delta t}$ (*blue*, *dashed*) for Greedy tolerance $\texttt{tol} = 0.001$. *Top line*: symmetric case $(\mu_1, 0, 0)$; *bottom*: non-symmetric $(0.5, \mu_2, 0.75)$. (**a**) Space-time, symmetric. (**b**) POD-Greedy, symmetric. (**c**) Space-time, non-symmetric. (**d**) POD-Greedy, non-symmetric

Work Load/Effort We now compare the computational effort as well as the storage amount (offline and online) of both variants, see Table 1.2.

Recall, that the chosen space-time method in the offline phase reduces to the Crank-Nicholson scheme. Hence, the offline complexities and storage requirements for the detailed solution of both variants indeed coincide (both linear in $\mathcal{N}_h$; we count $187 \approx 3\mathcal{N}_h$ elements). The detailed solution requires K solves (corresponding to the number of time steps) of a sparse system of size $\mathcal{N}_h$. However, the space-time precomputations needed to form the online system, rely on the full dimension $\mathcal{N}$.

In the online-phase, POD-Greedy needs to store (for the LTI case) Q_a reduced matrices and KQ_g vectors of size $N_{\text{POD-G}}$ for the time-dependent right-hand side. The RB solution amounts to solve K densely populated reduced systems. In the LTI case, one may reduce this to the computation of one LU-decomposition and then K triangular systems, i.e., $\mathcal{O}(N_{\text{POD-G}}^3 + KN_{\text{POD-G}}^2)$ operations.

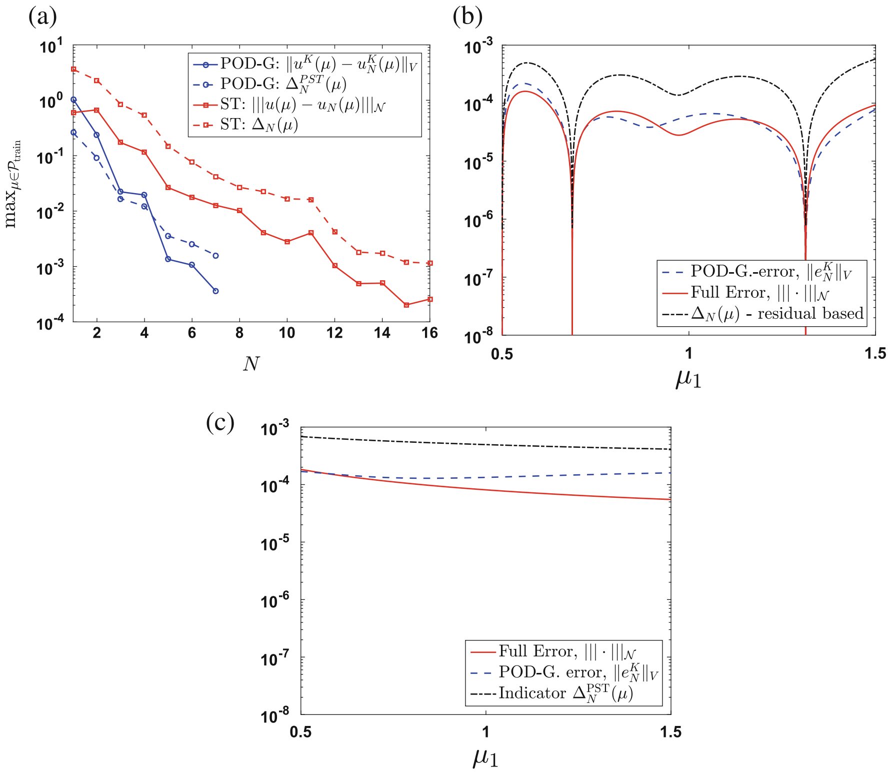


Fig. 1.3 Greedy error decay (**a**), space-time (**b**) and POD-Greedy (**c**) using the final time for POD-Greedy training and error measure on the symmetric test set

Table 1.2 Offline and online effort of the RBMs

	POD-Greedy		Space-time	
$\mathcal{O}()$ / #	Offline	Online	Offline	Online
Solution	$K\mathcal{N}_h$	$N_{\text{POD-G}}^3 + KN_{\text{POD-G}}^2$	$K\mathcal{N}_h$	N_{ST}^3
	4.032	2.520	4.032	1.728
Storage	$\sim 3\mathcal{N}_h$	$Q_aN_{\text{POD-G}}^2 + KQ_gN_{\text{POD-G}}$	$\sim 3\mathcal{N}_h$	$Q_b^2N_{\text{ST}}^2 + Q_bQ_fN_{\text{ST}}$
	187	528	187	2.352

We have $N_{\text{POD-G}} = 6$, $N_{\text{ST}} = 12$, $\mathcal{N}_h = 63$, $Q_a = Q_b = 4$, $Q_f = Q_g = 1$, $K = 64$; the corresponding numbers are given in the respective second row

The space-time method usually requires more storage, either by storing a suitable test space or—as described above—by the setup of the normal equations using the affine decomposition (1.10). The normal equations need memory of size $\mathcal{O}(Q_b^2N_{\text{ST}}^2 + Q_bQ_fN_{\text{ST}})$. The RB-solution requires the solution of one reduced linear system of dimension N_{ST}.

1.7 Conclusions

We have explained and compared both theoretically and numerically two different techniques to treat the time within the Reduced Basis Method (RBM). It is obvious that a fair and significant numerical comparison is a delicate task. In particular,

- different norms need to be considered;
- the choice of the error estimator/indicator within the POD-Greedy method has a significant impact;
- improving bounds for the involved constants will improve the results;
- we only consider such space-time methods that are equivalent to a time marching scheme (here Crank-Nicholson). If this is not the case, the offline cost for the space-time scheme will significantly increase;
- the number of time steps ($K = 64$) is moderate. As noted in Table 1.2, the online effort for POD-Greedy grows linearly with K, whereas the online space-time effort is independent of K. The reason is that the number of POD-Greedy basis functions stays the same and just more time steps are needed. The dimension of the space-time reduced system is independent of K. Increasing K (i.e., a higher temporal resolution or longer time horizons keeping Δt the same) will support space-time;
- we only consider LTI-systems. Both the reduction rate and the storage depend on this assumption.

Despite all these, we do think that our study indeed shows some facts, namely:

1. The POD-Greedy method allows one to use *any* time-marching scheme offline. Even if the space-time discretization is chosen in such a way that it coincides with a time-stepping scheme, this variant has an increased offline complexity. If the full space-time dimension is needed, one may have to resort to tensor product schemes [8].
2. In the online phase, the space-time method is more efficient concerning effort, whereas POD-Greedy uses less storage.
3. If a theoretical justification of an error bound in the full $\mathbb{X}$-norm is needed and Finite Elements shall be used, space-time seems to be the method of choice. If a Finite Volume discretization is chosen, POD-Greedy is more appropriate [5].

From these results, we tend to formulate the following recipe: If online computational time restrictions apply or for long-time horizons, the space-time approach is advisable even in those cases where it might cause an increased offline effort. If online storage restrictions apply or the use of a specific time-marching scheme is necessary, the POD-Greedy approach is advisable.

References

1. Andreev, R.: Stability of space-time Petrov-Galerkin discretizations for parabolic evolution equations. Ph.D. thesis, ETH Zürich, Nr. 20842 (2012)
2. Baur, U., Benner, P., Haasdonk, B., Himpe, C., Maier, I., Ohlberger, M.: Comparison of methods for parametric model order reduction of instationary problems. Preprint MPIMD/15-01, Max Planck Institute Magdeburg (2015)
3. Haasdonk, B.: Convergence rates of the POD-greedy method. ESAIM Math. Model. Numer. Anal. **47**(3), 859–873 (2013)
4. Haasdonk, B.: Reduced basis methods for parametrized PDEs – a tutorial introduction for stationary and instationary problems. In: Reduced Order Modelling. Luminy Book Series. IANS, University of Stuttgart, Stuttgart (2014)
5. Haasdonk, B., Ohlberger, M.: Reduced basis method for finite volume approximations of parametrized linear evolution equations. M2AN Math. Model. Numer. Anal. **42**(2), 277–302 (2008)
6. Hesthaven, J., Rozza, G., Stamm, B.: Certified reduced basis methods for parametrized partial differential equations. In: Springer Briefs in Mathematics. Springer, Cham (2016)
7. Huynh, D., Rozza, G., Sen, S., Patera, A.: A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. C.R. Math. Acad. Sci. Paris **345**(8), 473–478 (2007)
8. Mayerhofer, A.: Reduced basis methods for parabolic PDEs with parameter functions in high dimensions and applications in finance. Ph.D. thesis, Ulm University (2016)
9. Nochetto, R., Siebert, K., Veeser, A.: Theory of adaptive finite element methods: An introduction. In: DeVore, R., Kunoth A. (eds.) Multiscale, Nonlinear and Adaptive Approximation, pp. 409–542. Springer, Berlin (2009)
10. Quarteroni, A., Valli, A.: Numerical Approximation of Partial Differential Equations, vol. 23. Springer Science, Berlin (2008)
11. Quarteroni, A., Manzoni, A., Negri, F.: Reduced Basis Methods for Partial Differential Equations, vol. 92. Springer, Cham (2016)
12. Schwab, C., Stevenson, R.: Space-time adaptive wavelet methods for parabolic evolution problems. Math. Comput. **78**(267), 1293–1318 (2009)
13. Urban, K., Patera, A.: A new error bound for reduced basis approximation of parabolic partial differential equations. C.R. Math. Acad. Sci. Paris **350**(3–4), 203–207 (2012)
14. Urban, K., Patera, A.: An improved error bound for reduced basis approximation of linear parabolic problems. Math. Comput. **83**(288), 1599–1615 (2014)
15. Wloka, J.: Partial Differential Equations. Cambridge University Press, Cambridge (1987)

Chapter 2
Simultaneous Empirical Interpolation and Reduced Basis Method: Application to Non-linear Multi-Physics Problem

Cécile Daversin and Christophe Prud'homme

Abstract This paper focuses on the reduced basis method in the case of non-linear and non-affinely parametrized partial differential equations where affine decomposition is not obtained. In this context, Empirical Interpolation Method (EIM) (Barrault et al. C R Acad Sci Paris Ser I 339(9):667–672, 2004) is commonly used to recover the affine decomposition necessary to deploy the Reduced Basis (RB) methodology. The build of each EIM approximation requires many finite element solves which increases significantly the computational cost hence making it inefficient on large problems (Daversin et al. ESAIM proceedings, EDP Sciences, Paris, vol. 43, pp. 225–254, 2013). We propose a Simultaneous EIM and RB method (SER) whose principle is based on the use of reduced basis approximations into the EIM building step. The number of finite element solves required by SER can drop to $N + 1$ where N is the dimension of the RB approximation space, thus providing a huge computational gain. The SER method has already been introduced in Daversin and Prud'homme (C R Acad Sci Paris Ser I 353:1105–1109, 2015) through which it is illustrated on a 2D benchmark itself introduced in Grepl et al. (Modél Math Anal Numér 41(03):575–605, 2007). This paper develops the SER method with some variants and in particular a multilevel SER, SER(ℓ) which improves significantly SER at the cost of $\ell N + 1$ finite element solves. Finally we discuss these extensions on a 3D multi-physics problem.

2.1 Introduction

The demand in terms of real time simulations and uncertainty quantification is fast growing area in engineering, together with the size and the complexity of the considered problems. Reduced order methods, and in particular the reduced basis methods, play a critical role at breaking complexity.

C. Daversin (✉) • C. Prud'homme
Université de Strasbourg, CNRS, IRMA UMR 7501, 7 rue René Descartes, F-67000 Strasbourg, France
e-mail: daversin@math.unistra.fr; prudhomme@unistra.fr

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_2

Especially designed for real-time and many-query simulations, the Reduced Basis (RB) method offers an efficient evaluation of quantities of interest and covers a large range of problems among which non-affinely parametrized Partial Differential Equations (PDE). Based on the so-called offline/online decomposition, this method distinguishes the parameter independent terms whose computation is costly due to their dependence on the finite element dimension. Allowing to compute the latter only once, such a decomposition is not necessarily available in particular for non-affine/non-linear problems. The Empirical Interpolation Method (EIM) is classically used prior to the RB methodology to recover an affinely parametrized problem ensuring the availability of the offline/online decomposition. We proposed a simultaneous EIM-RB (SER) method [2] to circumvent the possibly dissuasive additional cost required by the EIM building step. Building together the affine decomposition as well as the RB approximation space, SER indeed requires only but a few finite elements solves.

Following up [2], this paper reports a finer analysis of SER and its variants along with its expanded use to non-linear multi-physics problem. In particular we introduce a multilevel SER, SER(ℓ) which improves significantly SER. After a reminder of the SER method in the context of non-affinely parametrized PDEs, we first give an overview of the investigated variants. All of them are based on the development of an error representation, initially designed to guide the construction of the RB approximation space from a Greedy algorithm. The second part illustrates these variants with results obtained on a benchmark introduced in [5] on which the SER preliminary results shown in [2] were based. The last part focuses on the application of SER and SER(ℓ) to a non-linear multi-physics problem from the HiFiMagnet project aiming to design an efficient model for high field magnets [4].

2.2 A Simultaneous EIM-RB Method

Let $u(\boldsymbol{\mu})$ be the solution of a non-linear and non-affinely parametrized PDE, where $\boldsymbol{\mu}$ denotes the p-vector of inputs defined in the parameter space $\mathscr{D} \subset \mathbb{R}^p$. The non-affine parametrization comes from the dependance of the PDE on at least one non-affine function $w(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})$. Considering $X \subset H^1(\Omega)$ an Hilbert space whose scalar product is denoted as $(\cdot,\cdot)_X$, the variational formulation of the PDE consists in finding $u(\boldsymbol{\mu}) \in X$ as a root of a functional r such that

$$r(u(\boldsymbol{\mu}), v; \boldsymbol{\mu}; w(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})) = 0 \; \forall v \in X. \tag{2.1}$$

2.2.1 Preliminaries

We denote by $X^{\mathscr{N}} \subset X$ the finite element approximation space of dimension $\mathscr{N}$ in which the approximation $u_{\mathscr{N}}(\boldsymbol{\mu})$ of $u(\boldsymbol{\mu})$ resides. The non-linearity of the considered PDE is handled through iterative methods. The following description relies on a Newton algorithm for which we introduce j the Jacobian associated with the functional r of (2.1), and ${}^k u(\boldsymbol{\mu})$ the solution at k-th Newton's iteration. The problem (2.1) then consists in finding $\delta^{k+1} u(\boldsymbol{\mu}) \equiv {}^{k+1}u(\boldsymbol{\mu}) - {}^k u(\boldsymbol{\mu}) \in X^{\mathcal{N}}$ such that

$$j(u, v; \boldsymbol{\mu}; {}^k u(\boldsymbol{\mu}); w({}^k u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu}))\delta^{k+1} u(\boldsymbol{\mu}) = -r({}^k u(\boldsymbol{\mu}), v; \boldsymbol{\mu}; w({}^k u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})) \tag{2.2}$$

2.2.1.1 Empirical Interpolation Method

The reduced basis method is based on an offline/online strategy assuming the existence of an affine decomposition of (2.2). The dependance of j and r on $w(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})$ stands in the way of the availability of such decomposition. In this context, the Empirical Interpolation method (EIM) is widely used to recover an affinely parametrized problem from (2.2) building an affine approximation $w_M(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})$ of $w(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})$ reading as

$$w_M(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu}) = \sum_{m=1}^{M} \beta_m^M(u(\boldsymbol{\mu}); \boldsymbol{\mu}) q_m(\boldsymbol{x}) \tag{2.3}$$

whose approximation coefficients β_m^M results from the resolution of a $M \times M$ system ensuring the exactness of w_M at a set of interpolation points $\{\boldsymbol{t}_i\}_{i=1}^M$.

To this end, we first introduce a subset Ξ of $\mathcal{D}$ from which a sample $\bar{S}_M = \{\bar{\boldsymbol{\mu}}_1, \ldots, \bar{\boldsymbol{\mu}}_M\} \in \mathcal{D}^M$ is built. The EIM approximation space $\bar{W}_M = span\{\bar{\boldsymbol{\xi}}_m \equiv w(u(\bar{\boldsymbol{\mu}}_m), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_m), 1 \leqslant m \leqslant M\}$ in which the approximation $w_M(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})$ shall reside consists of the set of w evaluations on $\bar{S}_M$ elements. As starting point of the EIM algorithm, the first sample point $\bar{\boldsymbol{\mu}}_1$ is picked in Ξ assuming $\bar{\boldsymbol{\xi}}_1 \neq 0$.

$$\bar{\boldsymbol{\xi}}_1 = w(u(\bar{\boldsymbol{\mu}}_1), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_1), \quad \boldsymbol{t}_1 = arg \sup_{\boldsymbol{x} \in \Omega} |\bar{\boldsymbol{\xi}}_1(\boldsymbol{x})|, \quad q_1 = \frac{\bar{\boldsymbol{\xi}}_1(\boldsymbol{x})}{\bar{\boldsymbol{\xi}}_1(\boldsymbol{t}_1)} \tag{2.4}$$

The next sample points $\{\bar{\boldsymbol{\mu}}_m\}_{m=2}^M$ are then selected through a Greedy algorithm as

$$\bar{\boldsymbol{\mu}}_M = arg \max_{\boldsymbol{\mu} \in \Xi} \inf_{z \in W_{M-1}} ||w(u(\boldsymbol{\mu}); .; \boldsymbol{\mu}) - z||_{L^\infty(\Omega)} \tag{2.5}$$

leading to the EIM approximation space enrichment $\bar{W}_M = \bar{W}_{M-1} \oplus span\{\bar{\boldsymbol{\xi}}_M\}$. The computation of the coefficients $\beta_m^{M-1}(u(\bar{\boldsymbol{\mu}}_M); \bar{\boldsymbol{\mu}}_M)$ allows to evaluate the residual $\boldsymbol{r}_M(\boldsymbol{x}) = w(u(\bar{\boldsymbol{\mu}}_M), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_M) - w_{M-1}(u(\bar{\boldsymbol{\mu}}_M), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_M)$ defining the interpolation point t_M and the next basis function $\boldsymbol{q}_M$ as

$$\boldsymbol{t}_M = arg \sup_{\boldsymbol{x} \in \Omega} |\boldsymbol{r}_M(\boldsymbol{x})|, \quad \boldsymbol{q}_M(\boldsymbol{x}) = \frac{\boldsymbol{r}_M(\boldsymbol{x})}{\boldsymbol{r}_M(\boldsymbol{t}_M)} \tag{2.6}$$

2.2.1.2 Reduced Basis Method

Defined as the linear combination of the finite element solutions forming the RB approximation space $W_N = span\{\boldsymbol{\xi}_i \equiv u_{\mathcal{N}}(\boldsymbol{\mu}_i), 1 \leqslant i \leqslant N\}$, the reduced basis approximation $u_N(\boldsymbol{\mu})$ of $u(\boldsymbol{\mu})$ reads as

$$u_N(\boldsymbol{\mu}) = \sum_{i=1}^{N} u_{N,i}(\boldsymbol{\mu})\boldsymbol{\xi}_i \tag{2.7}$$

Based on a sample $S_N = \{\boldsymbol{\mu}_1, \cdots, \boldsymbol{\mu}_N\} \subset \mathcal{D}$ with $N << \mathcal{N}$, W_N is built from the set of finite element solutions $\{u_{\mathcal{N}}(\boldsymbol{\mu}_i)\}_{i=1}^{N}$ which are orthonormalized, with respect to the scalar product of X, through a Gram-Schmidt algorithm. As $u_{\mathcal{N}}(\boldsymbol{\mu})$, the reduced basis approximation $u_N(\boldsymbol{\mu})$ has to satisfy the Eq. (2.1). The computation of the coefficients $u_{N,i}(\boldsymbol{\mu})$ in W_N (2.7) consists in solving the $N \times N$ reduced system (2.8), considering $\{\boldsymbol{\xi}_n\}_{i=1}^{N}$ as test functions

$$\sum_{j=1}^{N} j(\boldsymbol{\xi}_j, \boldsymbol{\xi}_l; \boldsymbol{\mu}; {}^k u_N; w({}^k u_N, \boldsymbol{x}; \boldsymbol{\mu}))\delta^{k+1}u_{N,j} = -r(\boldsymbol{\xi}_l; \boldsymbol{\mu}; {}^k u_N; w({}^k u_N, \boldsymbol{x}; \boldsymbol{\mu})) \tag{2.8}$$

where $\delta^{k+1}u_{N,j} \equiv ({}^{k+1}u_{N,j} - {}^k u_{N,j})$ and with $1 \leqslant l \leqslant N$.

2.2.2 SER *Method*

Offering both an efficient computation of W_N and an efficient assembly of the reduced system (2.8) by means of the precomputation of the parameter independent terms of (2.2), the affine decomposition obtained through the EIM approximation of $w(u(\boldsymbol{\mu}), \boldsymbol{x}; \boldsymbol{\mu})$ is a core enabler of the reduced basis method. The EIM initialization step (2.4) requires a single finite element solve to build the first EIM basis function. The building of the subsequent ones is based on a Greedy algorithm (2.5), for which the number of finite element solves required is proportional to the size of EIM trainset Ξ. The complexity scales both with the finite element dimension $\mathcal{N}$ and the size of Ξ, hence making the cost of the EIM offline step prohibitive.

Introduced in [2], the simultaneous EIM-RB (SER) method reduces this cost using the readily available reduced approximation into the Greedy algorithm instead of finite element solves. Due to the lack of reduced approximation at the initialization step (2.4), the finite element solve of the full non-linear problem (2.2) for the first EIM approximation w_1 cannot be avoided. But (2.2) is never solved afterwards. This leads to a first rough affine decomposition making the reduced basis methodology feasible, and it results in a first reduced approximation to be used in the EIM Greedy algorithm (2.5). From there, EIM and RB approximation spaces are enriched alternately using solely RB approximations making the number of finite element solves required drop to one for the EIM offline step.

The parameter selection process is then based on the last reduced basis approximation u_{M-1}

$$\bar{\boldsymbol{\mu}}_M = arg \max_{\boldsymbol{\mu}\in\Xi} \inf_{z\in W_{M-1}} ||w(u_{M-1}(\boldsymbol{\mu});.;\boldsymbol{\mu}) - z||_{L^\infty(\Omega)} \tag{2.9}$$

used as well to build the current EIM basis function

$$\bar{\boldsymbol{\xi}}_M = w\left(u_{M-1}(\bar{\boldsymbol{\mu}}_M);\boldsymbol{x};\bar{\boldsymbol{\mu}}_M\right). \tag{2.10}$$

The residual $\boldsymbol{r}_M(\boldsymbol{x}) = w(u_{M-1}(\bar{\boldsymbol{\mu}}_M),\boldsymbol{x};\bar{\boldsymbol{\mu}}_M) - w_{M-1}(u_{M-1}(\bar{\boldsymbol{\mu}}_M),\boldsymbol{x};\bar{\boldsymbol{\mu}}_M)$ is also computed from u_{M-1} giving t_M and q_M from (2.6).

2.2.2.1 Error Estimation

The Greedy algorithm used in the EIM offline step—either based on finite element solve (2.5) or on reduced basis approximation (2.9)—relies on an evaluation of the approximation error, defining a criterion for the parameter selection process. In the absence of such an error representation, the RB sample S_N on which the RB approximation space W_N is based is built from a random selection. This is the case for the preliminary results displayed in [2]. In order to improve the reliability of the reduced basis approximation, we introduce an error representation allowing to build W_N from a Greedy algorithm. In the context of SER, this should improve the quality of the reduced basis approximation used in the EIM algorithm and then the affine decomposition especially during the first steps of SER.

The definition of an error bound for non-linear but affinely parametrized problems is given in [6], based on the norm of the Riesz representation $\mathscr{Y}_r$ of the residual r such that $(\mathscr{Y}_r, v)_X = r(v)$.

The definition of such an error bound is not readily feasible for non-linear non affinely parametrized problems. Let us now introduce $r^{aff}_{N,M}$ as the evaluation of (2.1) from the reduced basis approximation u_N and the EIM approximation w_M which served to build it. The residual $r^{aff}_{N,M}(\boldsymbol{\mu})$ admits an affine decomposition composed of Q_r terms based on the coefficients $\beta^M_{q,m}$ and the basis functions $r_{q,m}$ coming from the EIM approximations (2.3).

$$\begin{aligned} r^{aff}_{N,M}(\boldsymbol{\mu}) &= r(u_N(\boldsymbol{\mu}), v; \boldsymbol{\mu}; w_M(u_N(\boldsymbol{\mu}),\boldsymbol{x};\boldsymbol{\mu})) \\ &= \sum_{q=1}^{Q_r}\sum_{m=1}^{M} \beta^M_{q,m}(u_N(\boldsymbol{\mu});\boldsymbol{\mu}) r_{q,m}(u_N(\boldsymbol{\mu}), v) \end{aligned} \tag{2.11}$$

The Riesz representations $\mathscr{Y}_{r_{q,m,n}}$ of $r_{q,m}(\xi_n, v)$ with $1 \leqslant q \leqslant Q_r$, $1 \leqslant m \leqslant M$, and $1 \leqslant n \leqslant N$ are computed offline, providing an efficient online evaluation of $\mathscr{Y}^{aff}_{rN,M}$. Inspired from [6] but not providing an error bound, the norm of the Riesz representation $\mathscr{Y}^{aff}_{rN,M}$ of $r^{aff}_{N,M}$ (2.11) provides an error representation.

We then use this representation to drive construction of S_N in the Greedy algorithm

$$\boldsymbol{\mu}_i = arg\ max_{\boldsymbol{\mu}\in\mathscr{D}} \parallel \mathscr{Y}^{aff}_{rN,M}(\cdot;\boldsymbol{\mu}) \parallel_X \tag{2.12}$$

2.2.2.2 Some SER Variants

Besides its use in the S_N building process (2.12) consisting in a first SER variant, the previously introduced error representation serves as a quantifier of the reduced basis approximation accuracy through the SER offline procedure. Various alternatives based on this error representation have been investigated, whose most relevant ones are detailed in the following. They are still illustrated from the results obtained on the 2D benchmark introduced in [5].

r-Adaptation

We remind that the SER methodology consists of the simultaneous enrichment of EIM and RB approximation spaces whose basis functions are alternately built one by one. A first alternative consists in changing the frequency of the affine decomposition updates, corresponding to perform the alternate build per groups of size r. In this context, $r = 1$ corresponds to the initial SER method while $r = M$ stands for the standard RB methodology. Intermediary stages with $1 < r < M$ were investigated in [2] with r constant for the whole offline step.

We propose to use error evaluations as a criterion, providing guidance to perform a smart adaptation of r during the SER process. Introduced as r-adaptation, this method is detailed in Algorithm 1.

The Greedy algorithm used in EIM (2.9) and in RB (2.12) offline stage selects the parameter which maximizes a representation of the approximation error. Based on the increment of this error representation between two updates, the r-adaptation method aims to continue the enrichment until a relevant decrease of the approximation error. This adaptation process distinguishes the update frequency r_{EIM} and r_{RB} of the EIM and RB approximation spaces.

Algorithm 1 *r*-adaptation method

```
for $i = m$ to $i = m + r_{EIM}$ do    ▷ Build $r_{EIM}$ EIM basis functions
    $\varepsilon_i \leftarrow \max_{\mu \in \Xi} \inf_{z \in W_{i-1}} ||w(u_{i-1}(\boldsymbol{\mu}); .; \boldsymbol{\mu}) - z||_{L^\infty(\Omega)}$ and $\bar{\boldsymbol{\mu}}_i \leftarrow arg \max_{\mu \in \Xi} \inf_{z \in W_{i-1}} ||w(u_{i-1}(\boldsymbol{\mu}); .; \boldsymbol{\mu}) - z||_{L^\infty(\Omega)}$
    Compute $\boldsymbol{r}_i(\boldsymbol{x}) = w(u_{i-1}(\bar{\boldsymbol{\mu}}_i), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_i) - w_{i-1}(u_{i-1}(\bar{\boldsymbol{\mu}}_i), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_i)$, deduce $\boldsymbol{t}_i$ and $q_i(\boldsymbol{x})$
    if $(\varepsilon_i - \varepsilon_{i-1})/\varepsilon_{i-1} < \mathrm{tol}_{EIM}$ then $r_{EIM} \leftarrow r_{EIM} + 1$ end if    ▷ Continue EIM approximation space enrichment
end for

for $j = n$ to $j = n + r_{RB}$ do    ▷ Build $r_{RB}$ RB basis functions
    $\varepsilon_i \leftarrow max_{\mu \in \mathscr{D}} \parallel \mathscr{Y}^{aff}_{r_{j-1},\, m+r_{EIM}}(\cdot; \boldsymbol{\mu}) \parallel_X$ and $\boldsymbol{\mu}_j \leftarrow arg\ max_{\mu \in \mathscr{D}} \parallel \mathscr{Y}^{aff}_{r_{j-1},\, m+r_{EIM}}(\cdot; \boldsymbol{\mu}) \parallel_X$
    $W_j \leftarrow W_{j-1} \oplus span\{\boldsymbol{\xi}_j \equiv u_{\mathscr{N}}(\boldsymbol{\mu}_j)\}$
    if $(\varepsilon_i - \varepsilon_{i-1})/\varepsilon_{i-1} < \mathrm{tol}_{RB}$ then $r_{RB} \leftarrow r_{RB} + 1$ end if    ▷ Continue RB approximation space enrichment
end for
```

Algorithm 2 Hybrid Greedy algorithm

```
for $i = m$ to $i = m + r_{EIM}$ do    ▷ Build $r_{EIM}$ EIM basis functions
    for $\mu \in \Xi$ do
        if $\parallel \mathscr{Y}^{aff}_{r_{n-1},\, i-1}(\boldsymbol{\mu}) \parallel_X / \max_{\mu \in \Xi} \parallel \mathscr{Y}^{aff}_{r_{n-1},\, i-1}(\boldsymbol{\mu}) \parallel_X < \mathrm{tol}$
        then $u(\boldsymbol{\mu}) \leftarrow u_N(\boldsymbol{\mu})$    ▷ EIM Greedy
        else $u(\boldsymbol{\mu}) \leftarrow u_{\mathscr{N}}(\boldsymbol{\mu})$ end if
    end for
    $\bar{\boldsymbol{\mu}}_i \leftarrow arg \max_{\mu \in \Xi} \inf_{z \in W_{i-1}} ||w(u(\boldsymbol{\mu}); .; \boldsymbol{\mu}) - z||_{L^\infty(\Omega)}$    ▷ employ $u_{\mathscr{N}}(\boldsymbol{\mu})$ or $u_N(\boldsymbol{\mu})$ depending on $\parallel \mathscr{Y}^{aff}_{r_{n-1},\, i-1}(\boldsymbol{\mu}) \parallel_X$
    Compute $\boldsymbol{r}_i(\boldsymbol{x}) = w(u(\bar{\boldsymbol{\mu}}_i), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_i) - w_{i-1}(u(\bar{\boldsymbol{\mu}}_i), \boldsymbol{x}; \bar{\boldsymbol{\mu}}_i)$, deduce $\boldsymbol{t}_i$ and $q_i(\boldsymbol{x})$
end for

for $j = n$ to $j = n + r_{RB}$ do    ▷ Build $r_{RB}$ RB basis functions
    $\boldsymbol{\mu}_j \leftarrow arg\ max_{\mu \in \mathscr{D}} \parallel \mathscr{Y}^{aff}_{r_{j-1},\, m+r_{EIM}}(\cdot; \boldsymbol{\mu}) \parallel_X$
    $W_j \leftarrow W_{j-1} \oplus span\{\boldsymbol{\xi}_j \equiv u_{\mathscr{N}}(\boldsymbol{\mu}_j)\}$
end for
```

Hybrid Greedy Algorithm

The accuracy of the reduced basis approximation plays a key role in the $\bar{S}_M$ building step especially for the first EIM basis functions. A reduced basis approximation of poor quality could then damage the EIM approximation and consequently the quality of the affine decomposition. We propose to assess the quality of the reduced basis approximation from the error representation used in (2.12) for each parameter of the trainset. As illustrated in Algorithm 2, the reduced basis approximation is employed solely on parameters for which it is considered as relevant. A parametric finite element solve based on the current affine decomposition is used for the remaining parameters to benefit from the precomputations while considering the whole trainset all through the SER offline step. This leads to an hybrid method combining the use of finite element and RB approximations within the EIM Greedy algorithm (2.5).

Multilevel SER(ℓ)

The last variant we propose rests on the application of the SER methodology several times during the offline step. The first level exactly corresponds to the previously introduced SER method whose Greedy algorithm employed for EIM is described in (2.9). Once the EIM and RB approximation spaces completed, the multilevel SER(ℓ) method consists of restarting the algorithm while benefiting from the reduced basis approximation coming from the previous level. Considering u_N^ℓ the reduced basis approximation obtained at the ℓ-th level, the EIM Greedy algorithm becomes

$$\bar{\boldsymbol{\mu}}_M = arg \max_{\boldsymbol{\mu}\in\Xi} \inf_{z\in W_{M-1}} ||w(u_N^{\ell-1}(\boldsymbol{\mu}); .; \boldsymbol{\mu}) - z||_{L^\infty(\Omega)} \tag{2.13}$$

Coming from the whole W_N approximation space, the reduced basis approximations used in the EIM Greedy algorithm from the second level offer EIM approximations of better quality. Based on this consideration, the resulting reduced basis approximations are expected to be more accurate as well.

2.3 Numerical Experiments

Based on the 2D non-linear and non affinely parametrized benchmark of Grepl et al. [5], the following numerical experiments illustrate the proposed SER variants. The results are presented with $M = N$ but this is not mandatory. We consider the $2D$ domain $\Omega =]0, 1[^2$ and the parameter space $\mathscr{D} = [0.01, 10]^2$, this problem consists in finding u such that

$$-\Delta u + \mu_1 \frac{e^{\mu_2 u} - 1}{\mu_2} = 100 \sin(2\pi x) \sin(2\pi y) \text{ in } \Omega \text{ with } \boldsymbol{\mu} = (\mu_1, \mu_2) \in \mathscr{D} \tag{2.14}$$

The non-affinely parametrization of the problem (2.14) resides in the function $g(u, x; \boldsymbol{\mu}) = \mu_1 \frac{e^{\mu_2 u}-1}{\mu_2}$. Based on a training set $\Xi \subset \mathscr{D}$ of size 225, its EIM approximation $g_M = \sum_{i=1}^{M} \beta_m^M(u, \boldsymbol{\mu}) q_m(\boldsymbol{x})$ allows to recover the required affine

Table 2.1 Impact of Greedy algorithm in W_N building

N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$
(a) Standard—RB random				*(b) Standard—RB Greedy*			
5	5	8.37e−3	6.33e−3	5	5	8.22e−3	6.27e−3
10	10	4.33e−4	2.10e−4	10	10	2.87e−4	2.09e−4
15	15	2.60e−4	1.44e−4	15	15	1.96e−5	1.47e−5
20	20	9.14e−5	4.69e−5	20	20	1.57e−5	1.32e−5
25	25	4.18e−5	1.15e−5	25	25	3.14e−6	2.52e−6

Table 2.2 Impact of Greedy algorithm in W_N building within SER

N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$
(a) SER*—RB random*				*(b)* SER*—RB Greedy*			
5	5	1.06e−2	8.23e−3	5	5	1.04e−2	8.09e−3
10	10	2.13e−3	1.6e−3	10	10	2.40e−3	1.87e−3
15	15	5.12e−4	4.22e−4	15	15	2.38e−4	2.01e−4
20	20	3.58e−5	2.27e−5	20	20	3.02e−5	1.67e−5
25	25	2.24e−5	1.44e−5	25	25	2.65e−5	1.94e−5

decomposition. We consider the absolute error on the solution u and on the output s defined as the average of u over Ω

$$\varepsilon^u_{M,N} = \parallel u_{\mathcal{N}} - u_N \parallel_{L_2} \qquad \varepsilon^s_{M,N} = \mid s_{\mathcal{N}} - s_N \mid \tag{2.15}$$

where $\cdot_{\mathcal{N}}$ is the finite element solution/output of the initial problem (2.1) and $\cdot_N$ the reduced basis solution/output.

The following tables display the maximum of the absolute errors (2.15) obtained on a set a 1000 realizations selected randomly in $\mathcal{D}$. Table 2.1b reproduces the results in [5] using the standard method with an RB approximation space built from (2.12), confirming the relevance of the proposed error representation.

Regarding the SER method, Table 2.2a displays the results coming from the use of a random selection process within the construction of the RB approximation space while the Table 2.2b illustrates the use of a Greedy algorithm in this context. Although slightly higher than the errors obtained in Table 2.1b, these errors come close the ones obtained in Table 2.1a for a reduced computational cost. Nevertheless, the impact of the previously introduced error representation on this application turns out to be limited.

Table 2.3 SER variants based on error representation

N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$
(a) r-adaptation (tol_{EIM}=20%, tol_{RB}=20%)				*(b) Hybrid EIM (tol = 20%)*			
5	5	1.04e−2	8.09e−3	5	5	1.03e−2	8.06e−3
10	10	2.40e−3	1.87e−3	10	10	2.29e−3	1.77e−3
15	15	2.34e−4	1.95e−4	15	15	2.25e−4	1.87e−4
20	20	3.46e−5	2.01e−5	20	20	3.08e−5	1.77e−5
25	25	1.61e−5	9.19e−6	25	25	1.95e−5	1.40e−5

Table 2.3a and b combine the previous Greedy algorithm used to build the RB approximation space with the first two SER variants. Table 2.3a focuses on the r-adaptation method (Algorithm 1) while Table 2.3b investigates the previously introduced ranking of parameters, depending on the reduced basis approximation reliability within the EIM Greedy algorithm (Algorithm 2). Compared to previous SER results displayed in Table 2.2, neither of these variants results in a significative improvement in the error. A similar behaviour is observed combining these two variants.

Table 2.4a–d display the results obtained with the multilevel SER(ℓ)) method. As expected, the results coming from the first level (Table 2.4a) are similar to ones of Table 2.4a. The observed disparity comes from the random selection process performed at RB offline stage which could result in a slightly different RB approximation space.

From the second level illustrated by Table 2.4b, we observe a significant decrease of the error which comes up to the standard case introduced in Table 2.1a. However, the error is no longer visibly evolving at the next levels as shown in Table 2.4c and d.

As to the results obtained with SER(ℓ) method combined with the use of the previous error representation through a Greedy algorithm within the RB offline stage, they are illustrated in Table 2.5a–d. Their analysis results in the same conclusion as for Table 2.4.

The previous analysis is supported by the convergence study of the considered EIM approximation illustrated in Fig. 2.1. The graph of Fig. 2.1 plots the maximal value of the functional used in the EIM Greedy algorithm depending on the number of basis functions which compose the EIM approximation space. Compared with the EIM convergence obtained using the standard RB methodology, we can indeed notice a significant error decrease from the second level. The next levels remains comparable to the latter, already very close to the EIM approximation performed from the standard method.

Table 2.4 SER(ℓ)—random

N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$
(a) SER(1)				*(b)* SER(2)				*(c)* SER(3)				*(d)* SER(4)			
5	5	1.06e−2	8.40e−3	5	5	9.13e−3	7.12e−3	5	5	7.26e−3	5.58e−3	5	5	8.67e−3	6.58e−3
10	10	2.33e−3	1.72e−3	10	10	3.19e−4	1.12e−4	10	10	2.00e−3	1.13e−3	10	10	5.07e−3	3.35e−3
15	15	6.51e−4	5.12e−4	15	15	7.56e−5	5.36e−5	15	15	5.50e−4	4.43e−4	15	15	2.78e−4	2.30e−4
20	20	2.32e−4	1.94e−4	20	20	1.54e−4	2.67e−5	20	20	2.08e−4	3.27e−5	20	20	2.67e−4	4.35e−5
25	25	7.08e−5	5.64e−5	25	25	3.52e−5	2.76e−5	25	25	1.37e−5	6.87e−6	25	25	5.62e−6	2.56e−6

Table 2.5 SER(ℓ)—RB Greedy

N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$	N	M	$\max(\varepsilon^u_{M,N})$	$\max(\varepsilon^s_{M,N})$
(a) SER(1)				*(b)* SER(2)				*(c)* SER(3)				*(d)* SER(4)			
5	5	1.04e−2	8.09e−3	5	5	9.17e−3	6.99e−3	5	5	7.93e−3	6.01e−3	5	5	8.46e−3	6.36e−3
10	10	2.39e−3	1.86e−3	10	10	2.89e−4	2.07e−4	10	10	2.99e−4	1.80e−4	10	10	4.34e−4	2.24e−4
15	15	2.38e−4	2.00e−4	15	15	4.12e−5	1.87e−5	15	15	1.75e−4	1.35e−4	15	15	6.28e−5	5.05e−5
20	20	3.03e−5	1.65e−5	20	20	1.44e−5	7.61e−6	20	20	1.69e−5	6.02e−6	20	20	1.76e−5	1.17e−5
25	25	3.42e−5	2.45e−5	25	25	2.72e−5	2.20e−5	25	25	7.86e−6	5.37e−6	25	25	1.92e−5	1.51e−5

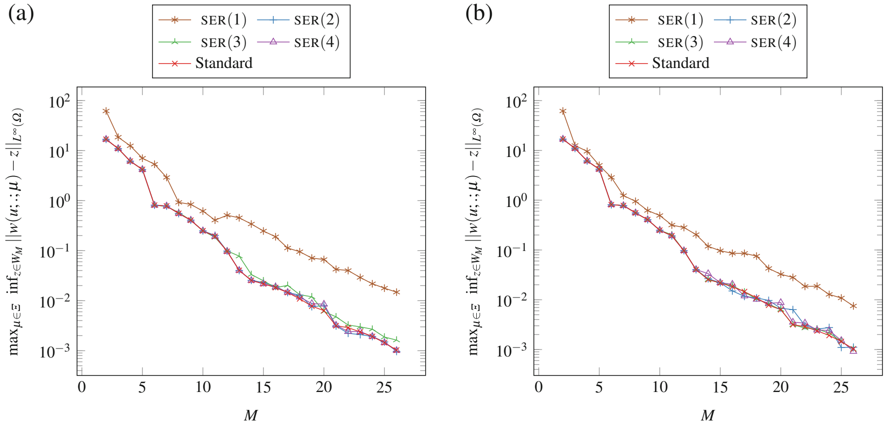


Fig. 2.1 SER(ℓ)—EIM convergence. (**a**) Random. (**b**) RB Greedy

2.4 Application to Multi-Physics Model

The HiFiMagnet project [4] aims at developing an efficient multi-physics model for high field magnets. We investigate the pertinence of the SER method on a 3D non-linear electro-thermal model. Considering the multilevel variant which has just been introduced, the initial SER method is denoted as SER(1) in the following.

2.4.1 Electro-Thermal Model

The considered electro-thermal model consists of the coupling of the electrical potential V in the magnet with the resulting temperature T. Thus, the temperature T is the solution of the non-linear coupled and non affinely parametrized thermo-electric problem

$$\begin{cases} -\nabla\cdot(\sigma(T)\nabla V) = 0 \\ -\nabla\cdot(k(T)\nabla T) = \sigma(T)\nabla V\cdot\nabla V \end{cases} \tag{2.16}$$

The non-linearity of (2.16) is coming from the dependance of the electrical (resp. thermal) conductivity $\sigma(T)$ (resp. $k(T)$) on temperature as well as the joule effect terms

$$\sigma(T) = \frac{\sigma_0}{1+\alpha(T-T_0)} \quad \text{and} \quad k(T) = \sigma(T)LT \tag{2.17}$$

The temperature coefficient α and the Lorentz number L are proper to the material, and σ_0 represents the electric conductivity at reference temperature $T = T_0$. Related to the current density j in the magnet, the current flow is modeled from a difference of electrical potential V_D between the current input and the current output imposed as Dirichlet boundary conditions. Other boundaries are considered as electrically insulated through a homogeneous Neumann condition.

$$\begin{cases} V = 0 \text{ on input, } V = V_D \text{ on output} \\ -\sigma(T)\nabla V \cdot \mathbf{n} = 0 \text{ on other boundaries} \end{cases} \tag{2.18}$$

The temperature increase due to the Joule effect is controlled with a water cooling of the magnet corresponding to a forced convection condition on the concerned regions, based on the water temperature T_w and on heat transfer coefficient h.

$$\begin{cases} -k(T)\nabla T \cdot \mathbf{n} = h(T - T_w) \text{ on cooled surfaces} \\ -k(T)\nabla T \cdot \mathbf{n} = 0 \text{ on other boundaries} \end{cases} \tag{2.19}$$

The input parameter $\boldsymbol{\mu} = (\sigma_0, \alpha, L, j, h, Tw) \in \mathbb{R}^6$ combines material properties and operating conditions parameters, while the considered output is the mean temperature over the magnet acting as a critical parameter in terms of magnet design. The definition of $\sigma(T)$ (2.17) from the input parameters thus makes the model (2.16) non-affinely parametrized as well as non-linear. The SER methodology readily applies in this context.

2.4.2 Application to Bitter Magnet

The first application focuses on the geometry illustrated in Fig. 2.2 whose mesh is composed of 15,388 nodes. This geometry stands for a sector of a Bitter magnet

Fig. 2.2 Temperature

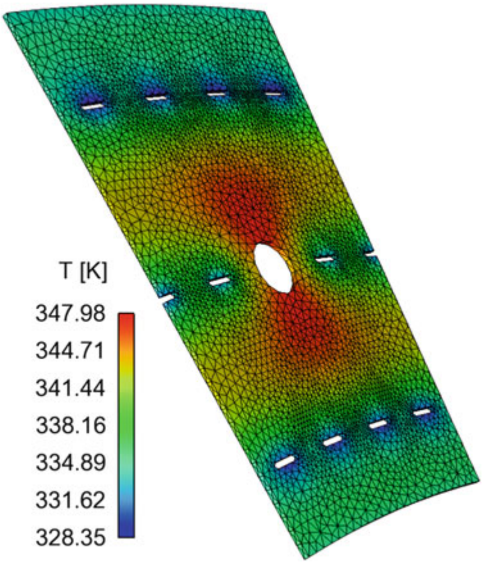


Table 2.6 Input parameters ranges

Input	Range
σ_0	$[40 \times 10^6, 60 \times 10^6]$ S m^{-1}
α	$[3.3 \times 10^{-3}, 3.5 \times 10^{-3}]$ K^{-1}
L	$[2.5 \times 10^{-8}, 2.9 \times 10^{-8}]$
j	$[30 \times 10^6, 60 \times 10^6]$ A m−2
h	$[50{,}000, 65{,}000]$ W m^{-2} K^{-1}
T_w	$[293, 313]$ K

Table 2.7 SER applied to electro-thermal model—Bitter magnet

N	M	$max(\varepsilon^u_{M,N})$	$max(\varepsilon^s_{M,N})$	N	M	$max(\varepsilon^u_{M,N})$	$max(\varepsilon^s_{M,N})$
(a) Standard—RB random				*(b)* SER(1)*—RB random*			
5	5	1.64e+1	1.94e−1	5	5	1.05e+1	2.09e−2
10	10	6.84e+0	8.24e−2	10	10	5.07e−1	3.42e−3
15	15	6.30e−2	4.90e−4	15	15	5.24e−1	1.05e−3
20	20	1.31e−2	1.65e−4	20	20	9.23e−2	1.89e−4
25	25	9.80e−3	6.74e−5	25	25	3.26e−2	1.90e−4

commonly used in the context of high field magnet facilities. The ranges considered for the input parameter given in Table 2.6 are chosen from physical considerations coming both from literature and experimental measures.

Performed in parallel on eight processors, the results given hereafter are based on a set of 1000 realizations for which parameters are randomly chosen in the ranges of Table 2.6. The non-linearity is handled by a fixed point iterative method, but this time with a Picard algorithm instead of a Newton method. Nevertheless, all previous considerations apply in the same way. As for the benchmark, we display the maximum of absolute errors (2.15) on solution and on output to be compared with reference results obtained with the standard method. The considered EIM trainset is of size 100.

The first Table 2.7a and b compare the SER[1] method with the standard one, both based on a randomly selection process to build the RB approximation space.

Turning to the previously introduced error estimator, Table 2.8 displays the errors obtained from its use into the parameter selection process. Besides its low impact on the 2D benchmark illustrated by Table 2.2a and b, the Greedy algorithm used to build the RB approximation space has a significant influence on this application. Indeed, it results in errors whose order of magnitude comes close to ones obtained with standard RB methodology.

The convergence study of the SER(1) method and its variants on this kind of application allows to go further in its analysis. To this purpose, Fig. 2.3a focuses on the convergence study related with the EIM approximation considered for the electrical conductivity $\sigma(T)$. Besides the expected error decrease, this plot highlights the enhancement coming from the use of the error representation in term

Table 2.8 SER(1)—RB Greedy

N	M	$max(\varepsilon^u_{M,N})$	$max(\varepsilon^s_{M,N})$
5	5	7.66e+0	2.71e−2
10	10	3.37e−1	7.82e−4
15	15	4.85e−2	2.64e−4
20	20	2.93e−2	3.40e−4
25	25	5.23e−3	4.59e−5

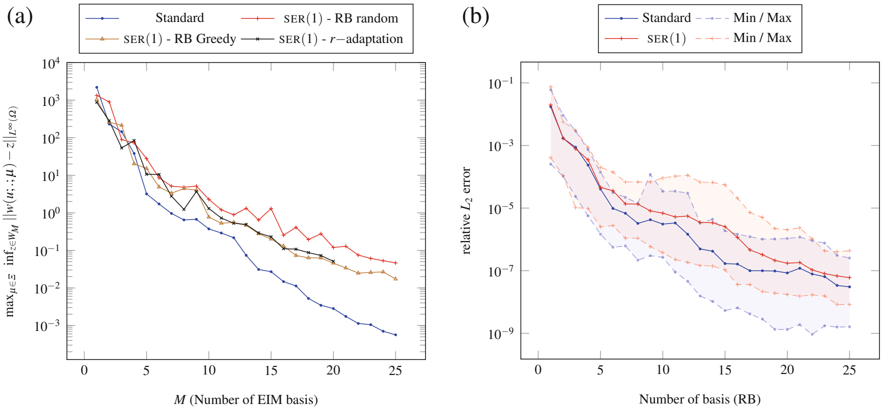


Fig. 2.3 SER method—EIM and RB convergence. (**a**) EIM approximation of $\sigma(T)$. (**b**) Reduced basis solution

of convergence. Figure 2.3b studies the relative L_2 error of the RB approximation depending on the number of basis functions. The resulting behaviour was expected as well.

Regarding its impact on the previous 2D benchmark, we investigate as well the use of multilevel SER(ℓ) method with $l \geqslant 1$ on the electro-thermal problem. In this context, Table 2.9 compares the results obtained at first and second levels. In spite of the random selection process, Table 2.9a is in good agreement with the previous Table 2.7b. Table 2.9b confirms the relevance of the multilevel variant since SER(2) already gives results which come close to those obtained with standard RB methodology.

As for the Fig. 2.1 in the case of the 2D benchmark, the convergence study of the considered EIM approximations allows to go further in the analysis. To this end, Fig. 2.4a (resp. Fig. 2.4b) compares the convergence resulting obtained from various level for the EIM approximation of $\sigma(T)$ (resp. $k(T)$). This study tends to confirm the preliminary results obtained on the benchmark which show pertinence of this SER variant.

Table 2.9 SER(ℓ)—random

N	M	$\max(\varepsilon^{u}_{M,N})$	$\max(\varepsilon^{s}_{M,N})$
(a) SER(1)			
5	5	2.56e+1	2.85e−1
10	10	9.55e+0	6.06e−2
(b) SER(2)			
5	5	7.78e+0	3.57e−2
10	10	2.09e+0	4.92e−3

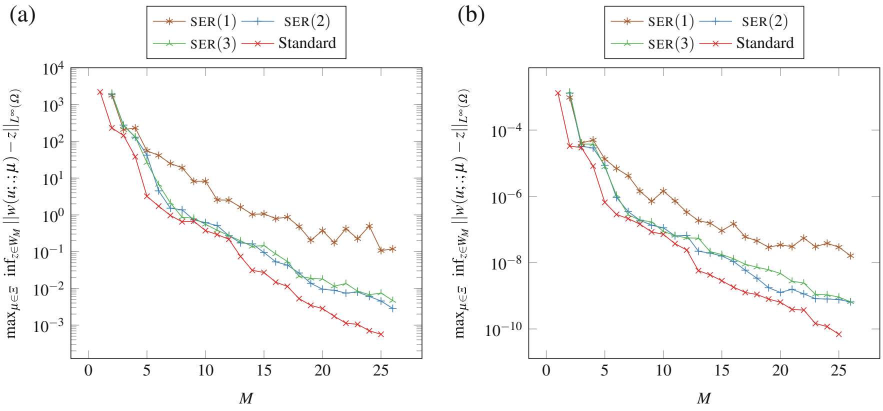


Fig. 2.4 SER(ℓ)—random—EIM convergence. (**a**) EIM $\sigma(T)$. (**b**) EIM $k(T)$

2.4.3 Application to Polyhelix Magnet

As an alternative of the previously mentioned Bitter magnets, the polyhelix magnets are designed to produce high magnetic fields. Detailed in [4], this technology rests on complex geometries leading to large problems in a numerical point of view. The saving in computational time offered by the SER method is thus all the more pertinent in this context. We propose to investigate its use on such a problem.

Based on a mesh composed of 2.2 millions of tetrahedra for approximatively 500,000 nodes (Fig. 2.5), the next simulations are performed on 12 processors. The computer used to perform this experiment is composed of two multi-threaded 6 cores CPUs and 141 GB of shared memory (Table 2.10).

The considered problem is similar to the one introduced in (2.16) for the Bitter magnet. The underlying non-linearity is handled by a Picard method with a given tolerance of 10^{-6}. Except the parameter related with the current flow, the input data are similar to the previous experiment. The current density previously considered is this time replaced by the difference of potential V_D which directly gives the Dirichlet boundary condition (2.18).

The next study is based on EIM and RB approximation spaces of size 5. It aims to compare the computational time necessary to perform the EIM Greedy algorithm

Fig. 2.5 Temperature

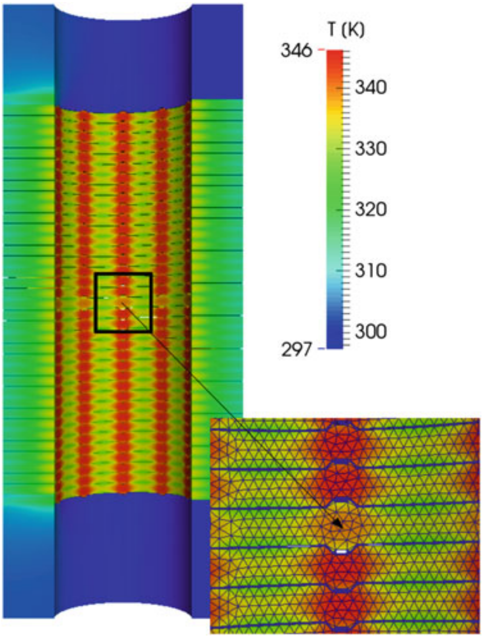


Table 2.10 Input parameters ranges

Input	Range
σ_0	$[50 \times 10^6, 50.2 \times 10^6]\,\mathrm{S\,m^{-1}}$
α	$[3.3 \times 10^{-3}, 3.5 \times 10^{-3}]\,\mathrm{K^{-1}}$
L	$[2.5 \times 10^{-8}, 2.9 \times 10^{-8}]$
V_D	$[55, 65]\,\mathrm{V}$
h	$[70{,}000, 90{,}000]\,\mathrm{W\,m^{-2}\,K^{-1}}$
T_w	$[293, 313]\,\mathrm{K}$

with the standard RB methodology and with the introduced SER method. To this end, we focus on the EIM approximation related with the electrical conductivity $\sigma(T)$ for which the considered trainset is composed of 100 parameters.

Consisting of the same finite element solve in both cases, the computational time related with the initialization of the EIM building step is approximatively 1760 s. The SER method uses the reduced basis approximation which results from the first EIM basis function while the standard methodology continue to use finite elements approximations. Regarding the first EIM Greedy algorithm for which the available reduced basis approximation rests on a single basis function, the mean time required to solve the reduced problem is 2.1 s compared with 2087 s for the corresponding finite element one. Regarding the whole set of resolutions performed within the EIM Greedy algorithm, this amounts to a factor close to 500 in terms of computational time.

Table 2.11 Performances of the SER method on a large scale multi-physics problem

N	Mean time per online realization (s)	Gain factor observed for EIM Greedy algorithm
1	2.1	495
2	4.3	321
3	7.7	213
4	6.1	254

Through each stage of the SER method, the EIM basis functions are built from a set of reduced basis approximations based on an enriched RB approximation space. In this context, Table 2.11 displays the mean time necessary for a single resolution and the resulting gain factor related with the whole EIM Greedy algorithm.

Still considering the mean temperature over the domain, the use of the SER method with $N = M = 5$ results in a maximal output error of $\varepsilon^s_{5,5} = 2.1 \times 10^{-1}$ compared to $\varepsilon^s_{5,5} = 4.5\times 10^{-2}$ with the standard methodology. We shall note that the initial version of the SER method—based on a random selection process regarding the building of the RB approximation space—shows convergence issues for some inputs. In our numerical experiments, SER combined with error representation provided an online code that was robust and did not fail to converge for some parameter set.

2.5 Conclusion

Intended as a follow-up of [2] introducing the SER method, this paper investigates some of its variants and in particular SER(ℓ) as well as the benefits of using an error representation to drive the SER process. We have proposed a methodology that can build simultaneously the affine decomposition of the original problem and the associated reduced basis model. Both (EIM and RB) are feeding each other. We recover convincing error convergence on EIM and RB on a standard benchmark problem as well as on 3D industrial nonlinear multi-physics application. In our numerical experiments, the SER methodology proved to be robust. We think that the proposed variants allow to improve this robustness by breaking premature stalls of our algorithm.

There are still various aspects of the methodology that can be investigated, we believe that SER opens various opportunities, and there is of course the theoretical question of the a priori convergence of the SER method—the methodology takes advantage of the underlying low-dimensional structures of the nonlinear PDE systems simultaneously from the EIM and RB sides.

Acknowledgements The authors would like to thank A.T. Patera (MIT) for the fruitful discussions we had as well as Vincent Chabannes (U. Strasbourg) and Christophe Trophime (CNRS). The authors acknowledge also the financial support of the ANR Chorus and the Labex IRMIA, and would like to thank PRACE for awarding us access to resource Curie based in France at CCRT as well as GENCI for awarding us access to resource Occigen based in France at Cines.

References

1. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. C.R. Acad. Sci. Paris Ser. I **339**(9), 667–672 (2004)
2. Daversin, C., Prud'homme, C.: Simultaneous empirical interpolation and reduced basis method for non-linear problems. C.R. Acad. Sci. Paris Ser. I **353**, 1105–1109 (2015)
3. Daversin, C., Veys, S., Trophime, C., Prud'homme, C.: A reduced basis framework: application to large scale non-linear multi-physics problems. In: ESAIM Proceedings, vol. 43, pp. 225–254. EDP Science, Paris (2013)
4. Daversin, C., Prud'homme, C., Trophime, C.: Full three-dimensional multiphysics model of high-field polyhelices magnets. IEEE Trans. Appl. Supercond. **26**(4) (2016). doi:10.1109/TASC.2016.2516241
5. Grepl, M.A., Maday Y., Nguyen, N.C. Patera, A.T.: Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. Modél. Math. Anal. Numér. **41**(03), 575–605 (2007)
6. Veroy, K., Prud'homme, C., Rovas, D.V, Patera, A.T.: A Posteriori Error Bounds for Reduced-Basis Approximation of Parametrized Noncoercive and Nonlinear Elliptic Partial Differential Equations. American Institute of Aeronautics and Astronautics Paper, 2003–3847 (2003)

Chapter 3
A Certified Reduced Basis Approach for Parametrized Optimal Control Problems with Two-Sided Control Constraints

Eduard Bader, Martin A. Grepl, and Karen Veroy

Abstract In this paper, we employ the reduced basis method for the efficient and reliable solution of parametrized optimal control problems governed by elliptic partial differential equations. We consider the standard linear-quadratic problem setting with distributed control *and* two-sided control constraints, which play an important role in many industrial and economical applications. For this problem class, we propose two different reduced basis approximations and associated error estimation procedures. In our first approach, we directly consider the resulting optimality system, introduce suitable reduced basis approximations for the state, adjoint, control, and Lagrange multipliers, and use a projection approach to bound the error in the reduced optimal control. For our second approach, we first reformulate the optimal control problem using two slack variables, we then develop a reduced basis approximation for both slack problems by suitably restricting the solution space, and derive error bounds for the slack based optimal control. We discuss benefits and drawbacks of both approaches and substantiate the comparison by presenting numerical results for a model problem.

3.1 Introduction

Optimal control problems governed by partial differential equations (PDEs) appear in a wide range of applications in science and engineering, such as heat phenomena, crystal growth, and fluid flow (see, e.g., [4, 8]). Their solution using classical discretization techniques such as finite elements (FE) or finite volumes can be

E. Bader (✉) • K. Veroy
Aachen Institute for Advanced Study in Computational Engineering Science (AICES), RWTH Aachen University, Schinkelstraße 2, 52062 Aachen, Germany
e-mail: bader@aices.rwth-aachen.de; veroy@aices.rwth-aachen.de

M.A. Grepl
Numerical Mathematics, RWTH Aachen University, Templergraben 55, 52056 Aachen, Germany
e-mail: grepl@igpm.rwth-aachen.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_3

computationally expensive and time-consuming. Often, additional parameters enter the problem, e.g., material or geometry parameters in a design exercise.

Previous work on reduced order methods for optimal control problems considered distributed but unconstrained controls or constrained but scalar controls. Elliptic optimal control problems with distributed control have been considered recently by Negri et al. [10]. The proposed error bound is based on the Banach-Nečas-Babuška (BNB) theory applied to the first order optimality system. The approach thus provides a combined bound for the error in the state, adjoint, and control variable, but it is only applicable to problems *without* control constraints. Since the bound requires the very costly computation of a lower bound to the inf-sup constant, Negri et al. [11] compute error estimates using a heuristic interpolant surrogate of that constant.

Based on the ideas in Tröltzsch and Volkwein [13], Kärcher and Grepl [6] proposed rigorous *and* online-efficient control error bounds for reduced basis (RB) approximations of scalar elliptic optimal control problems. These ideas are extended and improved in [7] to distributed control problems.

In a recent paper [1], we employed the RB method as a surrogate model for the solution of distributed *and* one-sided constrained optimal control problems governed by parametrized elliptic partial differential equations. In this paper we extend this work to two-sided control constraints. After stating the problem in Sect. 3.2 we present the following contributions:

- In Sect. 3.3 we extend previous work on reduced basis methods for variational inequalities in [1, 3] to the optimal control setting with two-sided control constraints. While we can derive an offline-online decomposable RB optimality system, we are only able to derive a *partially* offline-online decomposable control error bound that depends on the FE dimension of the control.
- In Sect. 3.4 we build on the recent work in [1, 14] and propose an RB slack approach for optimal control. We introduce two slack formulations for the optimal control problem, which we obtain by shifting the optimal control by each constraint. We are thus able to derive an offline-online decomposable RB optimality systems *and* control error bound. The evaluation of this bound is independent of the FE dimension of the problem, but requires the solution of three RB systems.

In Sect. 3.5 we propose a greedy sampling procedure to construct the RB spaces and in Sect. 3.6 we assess the properties of our methods by presenting numerical results for a Graetz flow problem.

3.2 General Problem Statement and Finite Element Discretization

In this section we introduce the parametrized linear-quadratic optimal control problem with elliptic PDE constraint and a constrained distributed control. We introduce a finite element (FE) discretization for the continuous problem and recall the first-order necessary (and in the convex setting sufficient) optimality conditions.

3.2.1 Preliminaries

Let $Y_{\rm e}$ with $H_0^1(\Omega) \subset Y_{\rm e} \subset H^1(\Omega)$ be a Hilbert space over the bounded Lipschitz domain $\Omega \subset \mathbb{R}^d, d \in \{1,2,3\}$, with boundary Γ.[1] The inner product and induced norm associated with $Y_{\rm e}$ are given by $(\cdot,\cdot)_Y$ and $\|\cdot\|_Y = \sqrt{(\cdot,\cdot)_Y}$. We assume that the norm $\|\cdot\|_Y$ is equivalent to the $H^1(\Omega)$-norm and denote the dual space of $Y_{\rm e}$ by $Y_{\rm e}'$. We also introduce the control Hilbert space $U_{\rm e} = L^2(\Omega)$, together with its inner product $(\cdot,\cdot)_U$, induced norm $\|\cdot\|_U = \sqrt{(\cdot,\cdot)_U}$, and associated dual space $U_{\rm e}'$.[2] Furthermore, let $\mathscr{D} \subset \mathbb{R}^P$ be a prescribed P-dimensional compact parameter set in which the P-tuple (input) parameter $\mu = (\mu_1, \dots, \mu_P)$ resides.

We directly consider a FE "truth" approximation for the exact infinite-dimensional optimal control problem. To this end, we define two conforming FE spaces $Y \subset Y_{\rm e}$ and $U \subset U_{\rm e}$ and denote their dimensions by $\mathscr{N}_Y = \dim(Y)$ and $\mathscr{N}_U = \dim(U)$. We assume that the truth spaces Y and U are sufficiently rich such that the FE solutions guarantee a desired accuracy over $\mathscr{D}$.

We next introduce the μ-dependent bilinear form $a(\cdot,\cdot;\mu) : Y \times Y \to \mathbb{R}$, and shall assume that $a(\cdot,\cdot;\mu)$ is (1) continuous for all $\mu \in \mathscr{D}$ with continuity constant $\gamma_a(\mu) < \infty$ and (2) coercive for all $\mu \in \mathscr{D}$ with coercivity constant $\alpha_a(\mu) > 0$. Furthermore, we introduce the μ-dependent continuous linear functional $f(\cdot;\mu) : Y \to \mathbb{R}$ and the bilinear form $b(\cdot,\cdot;\mu) : U \times Y \to \mathbb{R}$ with continuity constant $\gamma_b(\mu) < \infty$.

In anticipation of the optimal control problem, we introduce the parametrized control constraints $u_a(\mu), u_b(\mu) \in U$ and a desired state $y_d \in D$. Here, $D \subset L^2(\Omega_D)$ is a suitable FE space for the observation subdomain $\Omega_D \subset \Omega$. Furthermore, we note that the semi-norm $|\cdot|_D$ for $y \in L^2(\Omega)$ is defined by $|\cdot|_D = \|\cdot\|_{L^2(\Omega_D)}$.

The involved bilinear and linear forms as well as the control constraint are assumed to depend affinely on the parameter. For example we require for all $w, v \in Y$ and all parameters $\mu \in \mathscr{D}$ that $a(w,v;\mu) = \sum_{q=1}^{Q_a} \Theta_a^q(\mu)\, a^q(w,v)$ and $u_a(x;\mu) = \sum_{q=1}^{Q_{ua}} \Theta_{ua}^q(\mu)\, u_a^q(x)$ for some (preferably) small integers Q_a and Q_{ua}. Here, the coefficient functions $\Theta_\bullet^q(\cdot) : \mathscr{D} \to \mathbb{R}$ are continuous and depend on μ, whereas the continuous bilinear and linear forms, e.g., $a^q(\cdot,\cdot)$ and $u_a^q \in U$ do *not* depend on μ. Although we choose $y_d(x)$ to be parameter-independent, our approach directly extends to an affinely parameter-dependent $y_d(x;\mu)$ (see Kärcher et al. [7]).

[1] The subscript "e" denotes the "exact" infinite-dimensional continuous problem setting.

[2] The framework of this work directly extends to Neumann boundary controls $U_{\rm e} = L^2(\partial\Omega)$ or finite dimensional controls $U_{\rm e} = \mathbb{R}^m$. Also distributed controls on a subdomain $\Omega_U \subset \Omega$ or Neumann boundary controls on a boundary segment $\Gamma_U \subset \partial\Omega$ are possible.

For the development of a posteriori error bounds we also require additional ingredients. We assume that we are given a positive lower bound $\alpha_a^{\mathrm{LB}}(\mu) : \mathscr{D} \to \mathbb{R}_+$ for the coercivity constant $\alpha_a(\mu)$ of $a(\cdot,\cdot;\mu)$ such that $\alpha_a^{\mathrm{LB}}(\mu) \leq \alpha_a(\mu)\ \forall \mu \in \mathscr{D}$. Furthermore, we assume that we have upper bounds available for the constant $C_D^{\mathrm{UB}} \geq C_D = \sup_{w \in Y \setminus \{0\}} \frac{|w|_D}{\|w\|_Y} \geq 0\ \forall \mu \in \mathscr{D}$, and the continuity constant of the bilinear form $b(\cdot,\cdot;\mu)$: $\gamma_b^{\mathrm{UB}}(\mu) \geq \gamma_b(\mu)\ \forall \mu \in \mathscr{D}$. Here, the constant C_D depends on the parameter, since later we use $|\cdot|_D = \|\cdot\|_{L^2(\Omega_D(\mu))}$ (see Sect. 3.6). In our setting, it is possible to compute these constants (or their bounds) efficiently using an offline-online procedure (see [7, 12]).

3.2.2 Abstract Formulation of Linear-Quadratic Optimal Control Problems and the First-Order Optimality Conditions

We consider the following FE optimal control problem in weak form with $u_a(\mu) < u_b(\mu)$

$$\min_{\hat{y},\hat{u}} J(\hat{y},\hat{u}) = \frac{1}{2}|\hat{y} - y_d|_D^2 + \frac{\lambda}{2}\|\hat{u}\|_U^2, \qquad \lambda > 0 \tag{\textbf{P}}$$

$$\text{s.t.} \quad (\hat{y},\hat{u}) \in Y \times U \quad \text{solves} \quad a(\hat{y},\phi;\mu) = b(\hat{u},\phi;\mu) + f(\phi;\mu) \quad \forall \phi \in Y,$$
$$(u_a(\mu),\rho)_U \leq (\hat{u},\rho)_U \leq (u_b(\mu),\rho)_U \quad \forall \rho \in U^+,$$

where $U^+ := \{\rho \in U; \rho \geq 0 \text{ almost everywhere}\}$ and we dropped the μ-dependence of the state and control $(\hat{y},\hat{u})$ for the sake of readability. We note that the last line of (**P**) is equivalent to $\hat{u}$ being in the convex admissible set

$$U_{\mathrm{ad}} = \{\psi \in U; (u_a(\mu),\rho)_U \leq (\psi,\rho)_U \leq (u_b(\mu),\rho)_U\ \forall \rho \in U^+\}. \tag{3.1}$$

In the following we call problem (**P**) the "primal" problem, for which the existence and uniqueness of the solution is standard (see, e.g., [4]). The derivation of the necessary and sufficient first-order optimality system is straightforward: Given $\mu \in \mathscr{D}$, the optimal solution $(y,p,u,\sigma,\sigma_b) \in Y \times Y \times U \times U \times U$ satisfies

$$a(y,\phi;\mu) = b(u,\phi;\mu) + f(\phi;\mu) \qquad \forall \phi \in Y, \tag{3.2a}$$

$$a(\varphi,p;\mu) = (y_d - y,\varphi)_D \qquad \forall \varphi \in Y, \tag{3.2b}$$

$$(\lambda u,\psi)_U - b(\psi,p;\mu) = (\sigma,\psi)_U - (\sigma_b,\psi)_U \qquad \forall \psi \in U, \tag{3.2c}$$

$$(u_a(\mu) - u,\rho)_U \leq 0 \quad \forall \rho \in U^+, \qquad (u_a(\mu) - u,\sigma)_U = 0, \quad \sigma \geq 0, \tag{3.2d}$$

$$(u_b(\mu) - u,\rho)_U \geq 0 \quad \forall \rho \in U^+, \qquad (u_b(\mu) - u,\sigma_b)_U = 0, \quad \sigma_b \geq 0. \tag{3.2e}$$

Note that we follow a first-discretize-then-optimize approach here, for a more detailed discussion see [4, Sect. 3.2.4]).

In the following we comment on the FE-setting in this paper. We assume that the state variable is discretized by P_1, i.e., continuous and piecewise linear, and the control variable by P_0, i.e., piecewise constant finite elements. Next, we introduce two bases for the FE spaces Y and U, such that

$$Y = \operatorname{span}\{\phi_i^y, i = 1, \dots, \mathcal{N}_Y\} \quad \text{and} \quad U = \operatorname{span}\{\phi_i^u, i = 1, \dots, \mathcal{N}_U\},$$

where $\phi_i^y \geq 0$, $i = 1, \dots, \mathcal{N}_Y$, and $\phi_i^u \geq 0$, $i = 1, \dots, \mathcal{N}_U$, are the usual hat and bar basis functions. Using these basis functions we can express the functions $y, p \in Y$ and $u, \sigma, \sigma_b \in U$ as, e.g., $y = \sum_{i=1}^{\mathcal{N}_Y} y_i \phi_i^y$. The corresponding FE coefficient vectors are given by, e.g., $\underline{y} = (y_1, \dots, y_{\mathcal{N}_Y})^T \in \mathbb{R}^{\mathcal{N}_Y}$. Note that by definition of U^+ and since $\phi_i^u \geq 0$, the condition $\rho \in U^+$ in (3.2d) and (3.2e) translates into the condition $\underline{\rho} \geq 0$ for the corresponding coefficient vector. Further, we also introduce the control mass matrix M_U with entries $(M_U)_{ij} = (\phi_i^u, \phi_j^u)_U$, which is for a P_0 control discretization a positive diagonal matrix. Hence the point-wise and the 'weak' (averaged) constraint formulations are equivalent $u(x) \geq u_a(x; \mu) \Leftrightarrow (u, \rho)_U \geq (u_a(\mu), \rho)_U \ \forall \rho \in U^+$. However, this is in general not true for other control discretizations, e.g., P_1.

Based on the truth FE primal problem (**P**) we derive an RB primal problem (**P$_\mathbf{N}$**) and a rigorous a posteriori error bound for the error between the truth and RB control approximation in Theorem 1.

3.3 Reduced Basis Method for the Primal Problem

3.3.1 Reduced Basis Approximation

To begin, we define the RB spaces $Y_N \subset Y$, U_N, Σ_N, $\Sigma_{b,N} \subset U$ as well as the convex cones $\Sigma_N^+ \subset U^+$, $\Sigma_{b,N}^+ \subset U^+$ as follows: given N parameter samples $\mu^1, \dots, \mu^N$, we set

$$Y_N = \operatorname{span}\{\zeta_1^y, \dots, \zeta_{N_Y}^y\} = \operatorname{span}\{y(\mu^1), p(\mu^1), \dots, y(\mu^N), p(\mu^N)\}, \tag{3.3a}$$

$$U_N = \operatorname{span}\{\zeta_1^u, \dots, \zeta_{N_U}^u\} = \operatorname{span}\{u(\mu^1), \sigma(\mu^1), \sigma_b(\mu^1), \dots, u(\mu^N), \sigma(\mu^N), \sigma_b(\mu^N)\}, \tag{3.3b}$$

$$\Sigma_N = \operatorname{span}\{\zeta_1^\sigma, \dots, \zeta_{N_\sigma}^\sigma\} = \operatorname{span}\{\sigma(\mu^1), \dots, \sigma(\mu^N)\}, \tag{3.3c}$$

$$\Sigma_{b,N} = \operatorname{span}\{\zeta_1^{\sigma_b}, \dots, \zeta_{N_{\sigma_b}}^{\sigma_b}\} = \operatorname{span}\{\sigma_b(\mu^1), \dots, \sigma_b(\mu^N)\}, \tag{3.3d}$$

$$\Sigma_N^+ = \operatorname{span}_+\{\zeta_1^\sigma, \dots, \zeta_{N_\sigma}^\sigma\} \quad \text{and} \quad \Sigma_{b,N}^+ = \operatorname{span}_+\{\zeta_1^{\sigma_b}, \dots, \zeta_{N_{\sigma_b}}^{\sigma_b}\}, \tag{3.3e}$$

where we assume that the basis functions, $\zeta_1^\bullet, \dots \zeta_{N_\bullet}^\bullet$, are linearly independent and $\mathrm{span}_+\{\cdot\}$ indicates the cone spanned by non-negative combinations of the elements, i.e.

$$\mathrm{span}_+\{\zeta_1, \dots, \zeta_N\} = \left\{ \sum_{i=1}^{N} \alpha_i \zeta_i \,|\alpha_i \geq 0 \right\} .$$

Note that we employ integrated spaces for the state and adjoint as well as for the control (see Remarks 1 and 2). For the spaces Y_N and U_N we additionally assume that the basis functions are orthogonal, i.e., $(\zeta_i^y, \zeta_j^y)_Y = \delta_{ij}$ and $(\zeta_i^u, \zeta_j^u)_U = \delta_{ij}$, where δ_{ij} is the Kronecker delta. This orthogonality is favorable to keep the condition of the RB algebraic linear systems small [12]. In addition, we do not orthogonalize the basis ζ_i^σ, $\zeta_i^{\sigma_b}$ of the cones Σ_N^+, $\Sigma_{b,N}^+ \subset U^+$, because this non-negativity is used in the definition of the reduced problem ($\mathbf{P_N}$) of ($\mathbf{P}$).[3] Although the conditions $\zeta_i^\sigma \in \Sigma_N^+$ and $\zeta_i^{\sigma_b} \in \Sigma_{b,N}^+$ appear to be much more restrictive than ζ_i^σ, $\zeta_i^{\sigma_b} \in U^+$, we observe in numerical tests (not shown) that the RB approximations converge to the FE solutions with a similar rate as the control approximations. In addition, the RB approximations are comparable to the best possible approximations derived by projecting σ to Σ_N or Σ_N^+, analogously for σ_b. We describe the greedy sampling approach to construct the RB spaces in Sect. 3.5. Next, given the RB spaces in (3.3) we derive the RB primal problem

$$\min_{\hat{y}_N, \hat{u}_N} J(\hat{y}_N, \hat{u}_N) = \frac{1}{2}|\hat{y}_N - y_d|_D^2 + \frac{\lambda}{2}\|\hat{u}_N\|_U^2 \tag{$\mathbf{P_N}$}$$

$$\text{s.t. } (\hat{y}_N, \hat{u}_N) \in Y_N \times U_N \text{ solves } a(\hat{y}_N, \phi; \mu) = b(\hat{u}_N, \phi; \mu) + f(\phi; \mu) \; \forall \phi \in Y_N,$$

$$(u_a(\mu), \rho)_U \leq (\hat{u}_N, \rho)_U \; \forall \rho \in \Sigma_N^+, \; (u_b(\mu), \rho)_U \geq (\hat{u}_N, \rho)_U \; \forall \rho \in \Sigma_{b,N}^+.$$

The last line of ($\mathbf{P_N}$) defines the admissible set for u_N: $U_{\mathrm{ad},N} = \{\psi \in U_N; (u_a(\mu), \rho)_U \leq (\psi, \rho)_U \; \forall \rho \in \Sigma_N^+, \; (u_b(\mu), \rho)_U \geq (\psi, \rho)_U \; \forall \rho \in \Sigma_{b,N}^+\}$, which is in general *not* a subset of U_{ad} in (3.1). Analogously to the primal problem ($\mathbf{P}$) we obtain the RB optimality system: Given $\mu \in \mathscr{D}$, the optimal solution $(y_N, p_N, u_N, \sigma_N, \sigma_{b,N}) \in Y_N \times Y_N \times U_N \times \Sigma_N \times \Sigma_{b,N}$ satisfies

$$a(y_N, \phi; \mu) = b(u_N, \phi; \mu) + f(\phi; \mu) \qquad \forall \phi \in Y_N, \tag{3.4a}$$

$$a(\varphi, p_N; \mu) = (y_d - y_N, \varphi)_D \qquad \forall \varphi \in Y_N, \tag{3.4b}$$

$$(\lambda u_N, \psi)_U - b(\psi, p_N; \mu) = (\sigma_N, \psi)_U - (\sigma_{b,N}, \psi)_U \qquad \forall \psi \in U_N, \tag{3.4c}$$

$$(u_a(\mu) - u_N, \rho)_U \leq 0 \; \forall \rho \in \Sigma_N^+, \; (u_a(\mu) - u_N, \sigma_N)_U = 0, \qquad \sigma_N \in \Sigma_N^+, \tag{3.4d}$$

$$(u_b(\mu) - u_N, \rho)_U \geq 0 \; \forall \rho \in \Sigma_{b,N}^+, \; (u_b(\mu) - u_N, \sigma_{b,N})_U = 0, \qquad \sigma_{b,N} \in \Sigma_{b,N}^+. \tag{3.4e}$$

[3] Alternative methods to deal with the non-negativity can be found in [2].

Remark 1 (Existence, Uniqueness, Integrated Space Y_N) Since (**$\mathbf{P_N}$**) is a linear-quadratic optimal control problem over the closed convex admissible set $U_{\mathrm{ad},N}$, the existence and uniqueness of the RB optimal control u_N follows from standard arguments (see, e.g., [4, Theorem 1.43]). Also note that we use a single "integrated" reduced basis trial and test space Y_N for the state and adjoint equations as one ingredient to ensure stability of the system (3.4), see e.g. Kärcher [5].

Remark 2 (Stability, Integrated Space U_N) For the stability of the RB solutions we need to show that the RB inf-sup constants

$$\beta_N := \inf_{\psi_\sigma \in \Sigma_N} \sup_{\psi_u \in U_N} \frac{(\psi_\sigma, \psi_u)_U}{\|\psi_\sigma\|_U \|\psi_u\|_U}, \quad \beta_{b,N} := \inf_{\psi_{\sigma_b} \in \Sigma_{b,N}} \sup_{\psi_u \in U_N} \frac{(\psi_{\sigma_b}, \psi_u)_U}{\|\psi_{\sigma_b}\|_U \|\psi_u\|_U}$$

are bounded away from zero. We guarantee that $\beta_N, \beta_{b,N} \geq \beta > 0$ by enriching the RB control space with suitable supremizers [9]. Here, these supremizers are just the multiplier snapshots $\sigma(\mu^n), \sigma_b(\mu^n),\ 1 \leq n \leq N$; we thus have $\beta_N = \beta_{b,N} = \beta = 1$.

3.3.2 Primal Error Bound

We next propose an a posteriori error bound for the optimal control. The bound is based on [1], which uses an (1) RB approach for variational inequalities of the first kind [3], and (2) an RB approach for optimal control problems with a PDE constraint [7]. Before stating the main result, we define the following approximation errors (omitting μ-dependencies) of the RB primal system (3.4)

$$e_y = y - y_N, \quad e_p = p - p_N, \quad e_u = u - u_N, \quad e_\sigma = \sigma - \sigma_N, \quad e_{\sigma_b} = \sigma_b - \sigma_{b,N},$$

as well as the residuals in the next definition.

Definition 1 (Residuals) The residuals of the state equation, the adjoint equation w.r.t. (3.2a)–(3.2c) are defined for all $\mu \in \mathscr{D}$ by

$$r_y(\phi; \mu) = b(u_N, \phi; \mu) + f(\phi; \mu) - a(y_N, \phi; \mu) \qquad \forall \phi \in Y, \tag{3.5a}$$

$$r_p(\varphi; \mu) = (y_d - y_N, \varphi)_D - a(\varphi, p_N; \mu) \qquad \forall \varphi \in Y, \tag{3.5b}$$

$$r_u(\psi; \mu) = -\lambda(u_N, \psi)_U + b(\psi, p_N; \mu) + (\sigma_N, \psi)_U - (\sigma_{b,N}, \psi)_U \qquad \forall \psi \in U. \tag{3.5c}$$

Theorem 1 (Primal Error Bound) *Let u and u_N be the optimal controls of the FE primal problem* (**P**) *and of the RB primal problem* (**$\mathbf{P_N}$**)*, respectively. Then the error* *in the optimal control satisfies for any given parameter* $\mu \in \mathscr{D}$

$$\|e_u\|_U \le \Delta_N^{\mathrm{pr}}(\mu),$$

where $\Delta_N^{\mathrm{pr}}(\mu) := c_1(\mu) + \sqrt{c_1(\mu)^2 + c_2(\mu)}$ *with nonnegative coefficients*

$$c_1(\mu) = \frac{1}{2\lambda}\left(\|r_u\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p\|_{Y'} + \lambda(\delta_1 + \delta_{1b})\right), \tag{3.6a}$$

$$\begin{aligned} c_2(\mu) = \frac{1}{\lambda}\Bigg[&\frac{2}{\alpha_a^{\mathrm{LB}}}\|r_y\|_{Y'}\|r_p\|_{Y'} + \frac{1}{4}\left(\frac{C_D^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\left(\|r_y\|_{Y'} + \gamma_b^{\mathrm{UB}}(\delta_1 + \delta_{1b})\right)\right)^2 \\ &+ \left(\|r_u\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p\|_{Y'} + \sqrt{2(\sigma_N, \sigma_{b,N})_U}\right)(\delta_1 + \delta_{1b}) + \delta_2 + \delta_{2b}\Bigg], \end{aligned} \tag{3.6b}$$

and $\delta_1 = \|[u_a - u_N]_+\|_U$, $\delta_2 = ([u_a - u_N]_+, \sigma_N)_U$, $\delta_{1b} = \|[u_N - u_b]_+\|_U$, $\delta_{2b} = ([u_N - u_b]_+, \sigma_{b,N})_U$.
Here $[\cdot]_+ = \max(\cdot, 0)$ denotes the positive part (a.e.). Note that we sometimes use $r_\bullet$ instead of $r_\bullet(\cdot;\mu)$ and omit the μ-dependencies on the r.h.s. of (3.6) for a better readability.

Proof This proof follows the proof of the primal error bound from [1]. Since the FE optimal solution $(y, p, u, \sigma, \sigma_b)$ satisfies the optimality conditions (3.2), we obtain the following error-residual equations:

$$a(e_y, \phi; \mu) - b(e_u, \phi; \mu) = r_y(\phi; \mu) \quad \forall \phi \in Y, \tag{3.7a}$$

$$a(\varphi, e_p; \mu) + (e_y, \varphi)_D = r_p(\varphi; \mu) \quad \forall \varphi \in Y, \tag{3.7b}$$

$$\lambda(e_u, \psi)_U - b(\psi, e_p; \mu) - (e_\sigma, \psi)_U + (e_{\sigma_b}, \psi)_U = r_u(\psi; \mu) \quad \forall \psi \in U. \tag{3.7c}$$

From (3.7a) with $\phi = e_y$, (3.7b) with $\varphi = e_p$, and $\alpha_a^{\mathrm{LB}}(\mu) \le \alpha_a(\mu)$ we infer that

$$\|e_y\|_Y \le \frac{1}{\alpha_a^{\mathrm{LB}}}\left(\|r_y\|_{Y'} + \gamma_b^{\mathrm{UB}}\|e_u\|_U\right), \quad \|e_p\|_Y \le \frac{1}{\alpha_a^{\mathrm{LB}}}\left(\|r_p\|_{Y'} + C_D^{\mathrm{UB}}|e_y|_D\right). \tag{3.8}$$

Choosing $\phi = e_p$, $\varphi = e_y$, $\psi = e_u$ in (3.7), adding (3.7b) and (3.7c), and subtracting (3.7a) results in

$$\lambda\|e_u\|_U^2 + |e_y|_D^2 \le \|r_y\|_{Y'}\|e_p\|_Y + \|r_p\|_{Y'}\|e_y\|_Y + \|r_u\|_{U'}\|e_u\|_U + (e_\sigma - e_{\sigma_b}, e_u)_U. \tag{3.9}$$

Next, we bound $(e_\sigma, e_u)_U$ and $-(e_{\sigma_b}, e_u)_U$. We first consider $(e_\sigma, e_u)_U$ and note that

$$\begin{aligned}(e_\sigma, e_u)_U &= (\sigma - \sigma_N, u - u_N)_U = (\sigma, u - u_N)_U + (\sigma_N, u_N - u)_U \\ &= (\sigma, u - u_a(\mu))_U + (\sigma, u_a(\mu) - u_N)_U + (\sigma_N, u_N - u_a(\mu))_U + (\sigma_N, u_a(\mu) - u)_U,\end{aligned}$$

where, except for the second term, all terms are nonpositive, see (3.2d) and (3.4d). Hence

$$\begin{aligned}(e_\sigma, e_u)_U &\le (\sigma, u_a(\mu) - u_N)_U \le (\sigma, [u_a(\mu) - u_N]_+)_U \\ &= (\sigma - \sigma_N, [u_a(\mu) - u_N]_+)_U + (\sigma_N, [u_a(\mu) - u_N]_+)_U \le \|e_\sigma\|_U \;\delta_1 + \delta_2.\end{aligned} \tag{3.10}$$

Analogously, we bound $-(e_{\sigma_b}, e_u)_U \le \|e_{\sigma_b}\|_U \;\delta_{1b} + \delta_{2b}$. Most significantly, it remains to bound the terms $\|e_\sigma\|_U$ and $\|e_{\sigma_b}\|_U$, which we achieve in two steps: First, we relate $\|e_\sigma\|_U$ and $\|e_{\sigma_b}\|_U$ with $\|e_\sigma - e_{\sigma_b}\|_U$ by

$$\|e_\sigma\|_U^2 + \|e_{\sigma_b}\|_U^2 = \|e_\sigma - e_{\sigma_b}\|_U^2 + 2\big((\sigma, \sigma_b)_U - (\sigma_N, \sigma_b)_U + (\sigma_N, \sigma_{b,N})_U - (\sigma, \sigma_{b,N})_U\big)$$

If we employ $(\sigma, \sigma_b)_U = 0$, $(\sigma_N, \sigma_b)_U \ge 0$, and $(\sigma, \sigma_{b,N})_U \ge 0$, we obtain

$$\|e_\sigma\|_U, \|e_{\sigma_b}\|_U \le \|e_\sigma - e_{\sigma_b}\|_U + \sqrt{2(\sigma_N, \sigma_{b,N})_U}. \tag{3.11}$$

Second, we focus on the optimality residual (3.7c), use the inf-sup stability of $(\cdot, \cdot)_U$ and (3.8) to derive $\|e_\sigma - e_{\sigma_b}\|_U \le \|r_u\|_{U'} + \lambda\|e_u\|_U + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\left(\|r_p\|_{Y'} + C_D^{\mathrm{UB}}|e_y|_D\right)$. Next, we employ the inequalities (3.8) and (3.11) in (3.9) to obtain

$$\begin{aligned}\lambda\|e_u\|_U^2 + |e_y|_D^2 \le\; & \|e_u\|_U\left(\|r_u\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p\|_{Y'} + \lambda(\delta_1 + \delta_{1b})\right) \\ &+ \frac{2}{\alpha_a^{\mathrm{LB}}}\|r_y\|_{Y'}\|r_p\|_{Y'} + |e_y|_D\frac{C_D^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\left(\|r_y\|_{Y'} + \gamma_b^{\mathrm{UB}}(\delta_1 + \delta_{1b})\right) \\ &+ \left(\|r_u\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p\|_{Y'} + \sqrt{2(\sigma_N, \sigma_{b,N})_U}\right)(\delta_1 + \delta_{1b}) + \delta_2 + \delta_{2b}.\end{aligned} \tag{3.12}$$

It thus follows from applying Young's inequality to the $|e_y|_D$-terms in (3.12) that

$$\|e_u\|_U^2 - 2c_1(\mu)\|e_u\|_U - c_2(\mu) \le 0,$$

where $c_1(\mu)$ and $c_2(\mu)$ are given in (3.6). Solving the last inequality for the larger root yields $\|e_u\|_U \le c_1(\mu) + \sqrt{c_1(\mu)^2 + c_2(\mu)} = \Delta_N^{\mathrm{pr}}(\mu)$.

We note that most of the ingredients of the primal error bound $\Delta_N^{\mathrm{pr}}(\mu)$ introduced in Theorem 1 are standard, i.e., the dual norms of state, adjoint, and control residuals, as well as coercivity and continuity constants or rather their lower and upper bounds [7, 12]. The only non-standard terms are $\delta_1, \delta_2, \delta_{1b}$ and δ_{2b}, which measure the constraint-violation of the RB optimal control u_N. As a result, the online computational cost to evaluate $\delta_\bullet$—and hence the error bound $\Delta_N^{\mathrm{pr}}(\mu)$—depends on the FE control dimension $\mathscr{N}_U$, requiring $\mathscr{O}((N_U + N_\sigma + N_{\sigma_b})\mathscr{N}_U)$ operations.

3.4 Slack Problem and the Primal-Slack Error Bound

In this section we introduce a reformulation of the original primal problem by means of a slack variable. We extend the ideas presented for the one-sided control-constrained problem in [1] to the two-sided control-constrained problem. First, we reformulate the original optimization problem (**P**) by replacing the control variable with a slack variable that depends on one of the two constraints $u_a(\mu)$ or $u_b(\mu)$. Second, we use snapshots of the slack variable to construct an associated convex cone, leading to strictly feasible approximations w.r.t. either the lower or upper constraint. Third, we derive two RB slack problems by restricting the RB-slack coefficients to a convex cone. And finally, we propose an a posteriori $\mathscr{N}$-independent error bound for RB slack approximation w.r.t. either the lower or upper constraint in Theorem 2.

3.4.1 FE and RB Slack Problem

We consider the FE optimization problem (**P**) and introduce the slack variable $s \in U^+$ given by

$$s = u - u_a(\mu) \tag{3.13}$$

together with the corresponding FE coefficient vector $\underline{s} = \underline{u} - \underline{u}_a(\mu)$, where we state the slack variable w.r.t. $u_a(\mu)$. Here, we again omit the explicit dependence of u and s on the parameter μ. We note that, by construction, the feasibility of u w.r.t. $u_a(\mu)$ is equivalent to $M_U \underline{s} \geq 0$, which in turn is equivalent to $\underline{s} \geq 0$, if we are using P_0 elements.

If we substitute u by $s + u_a(\mu)$ in (**P**), we obtain the "slack" optimization problem

$$\min_{\hat{y},\hat{s}} J_s(\hat{y},\hat{s}) = \frac{1}{2}|\hat{y} - y_d|_D^2 + \frac{\lambda}{2}\|\hat{s} + u_a(\mu)\|_U^2 \tag{S}$$

$$\text{s.t.} \quad (\hat{y},\hat{s}) \in Y \times U^+ \quad \text{solves} \quad a(\hat{y},\phi;\mu) = b(\hat{s} + u_a(\mu),\phi;\mu) + f(\phi;\mu) \; \forall \phi \in Y,$$

$$(u_b(\mu),\rho)_U \geq (\hat{s} + u_a(\mu),\rho)_U \quad \forall \rho \in U^+.$$

Analogously, we define a slack variable $s_b = u_b(\mu) - u$ w.r.t. $u_b(\mu)$ and recast (**P**) w.r.t. s_b. We do not state this minimization problem explicitly since it is analogous to (**S**).

In the following we derive two RB slack problems w.r.t. $u_a(\mu)$ and $u_b(\mu)$. We start with the former and reuse the RB space Y_N, introduced in Sect. 3.3.1, for the state and adjoint variables. Furthermore, for the RB approximation of the slack variable s we simply introduce an RB slack space S_N and a convex cone S_N^+ by shifting the control snapshots of (**P**) with the control constraint $u_a(\mu)$

$$S_N = \operatorname{span}\{\, \zeta_1^s, \dots, \zeta_{N_S}^s \,\} = \operatorname{span}\{\, u(\mu^1) - u_a(\mu^1), \dots, u(\mu^N) - u_a(\mu^N) \,\}, \tag{3.14a}$$

$$S_N^+ = \operatorname{span}_+\{\, \zeta_1^s, \dots, \zeta_{N_S}^s \,\} \subset U^+. \tag{3.14b}$$

We assume that the snapshots $\zeta_1^s, \dots, \zeta_{N_s}^s$ are linearly independent and not orthogonalized. Further, we need to consider a Lagrange multiplier $\sigma_b^s \in U^+$ for the constraint $u_b(\mu)$ by incorporating the RB space $\Sigma_{b,N} \subset U$, as well as the convex cone $\Sigma_{b,N}^+$ from (3.3d) and (3.3e). Overall, for an RB approximation $s_N \in S_N^+ \subset U^+$ of s, we have $s_N \geq 0$. From the definition of the slack variable $s = u - u_a(\mu)$, see (3.13), we derive the control approximation $u^s := s_N + u_a(\mu)$ that satisfies $u^s \geq u_a(\mu)$. However, we can not conclude $u^s \leq u_b(\mu)$ since the slack approximation s_N is constructed—as the slack variable s in (3.13)—using information from $u_a(\mu)$ but not $u_b(\mu)$.

Overall, employing the RB spaces in (**S**) results in the RB slack problem

$$\min_{\hat{y}_N^s, \hat{s}_N} J_s(\hat{y}_N^s, \hat{s}_N) = \frac{1}{2}|\hat{y}_N^s - y_d|_D^2 + \frac{\lambda}{2}\|\hat{s}_N + u_a(\mu)\|_U^2 \tag{$\mathbf{S_N}$}$$

$$\text{s.t. } (\hat{y}_N^s, \hat{s}_N) \in Y_N \times S_N^+ \quad \text{solves} \quad a(\hat{y}_N^s, \phi; \mu) = b(\hat{s}_N + u_a(\mu), \phi; \mu) + f(\phi; \mu) \ \forall \phi \in Y_N,$$

$$(u_b(\mu), \rho)_U \geq (\hat{s}_N + u_a(\mu), \rho)_U \quad \forall \rho \in \Sigma_{b,N}^+,$$

As in the RB primal problem ($\mathbf{P_N}$), the existence and uniqueness of the RB optimal control follows from the same arguments as in the Remark 1. Next, we derive the optimality conditions for s_N; however, we here follow the 'first-discretize-then-optimize' approach that will eventually lead to a feasible—w.r.t. $u_a(\mu)$—approximation of the control. We perform two steps.

First, we use the RB-representations of y_N^s, s_N with their RB-coefficient vectors $\underline{y}_N^s, \underline{s}_N$ to discretize ($\mathbf{S_N}$). Since the algebraic RB slack problem is simple to derive, we only state the main crucial condition that $\underline{s}_N \geq 0$. Next, we derive the first-order optimality conditions. We introduce a discrete Lagrange multiplier $\underline{\hat{\omega}}_N \in \mathbb{R}^{N_S}$, $\underline{\hat{\omega}}_N \geq 0$, ensuring the non-negativeness of $\underline{\hat{s}}_N$ and derive the following necessary (and here sufficient) first-order optimality system: Given $\mu \in \mathscr{D}$, the optimal RB slack solution coefficients $(\underline{y}_N^s, \underline{s}_N, \underline{p}_N^s, \underline{\sigma}_{b,N}^s, \underline{\omega}_N) \in \mathbb{R}^{N_Y} \times \mathbb{R}^{N_Y} \times \mathbb{R}^{N_S} \times$

$\mathbb{R}^{N_{\sigma_b}} \times \mathbb{R}^{N_S}$ satisfy (omitting all μ-dependencies)

$$A_N \underline{y}_N^s = F_N + B_N^s \underline{s}_N + B_{a,N}^s, \tag{3.15a}$$

$$A_N^T \underline{p}_N^s = Y_{d,N} - D_N \underline{y}_N^s, \tag{3.15b}$$

$$\lambda U_N^s \underline{s}_N + \lambda U_{a,N}^s - (B_N^s)^T \underline{p}_N^s = \underline{\omega}_N - U_N^{\sigma_b,s} \underline{\sigma}_{b,N}^s, \tag{3.15c}$$

$$\underline{s}_N{}^T \underline{\omega}_N = 0, \quad \underline{s}_N \geq 0, \quad \underline{\omega}_N \geq 0, \tag{3.15d}$$

$$(U_{b,N}^{\sigma_b} - U_{a,N}^{\sigma_b} - U_N^{\sigma_b,s} \underline{s}_N)^T \underline{\sigma}_{b,N}^s = 0, \quad U_{b,N}^{\sigma_b} - U_{a,N}^{\sigma_b} \geq U_N^{\sigma_b,s} \underline{s}_N, \quad \underline{\sigma}_{b,N}^s \geq 0. \tag{3.15e}$$

where the reduced basis matrices and vectors are given by

$$(A_N)_{ij} = a(\zeta_i^y, \zeta_j^y), \quad (F_N)_i = f(\zeta_i^y), \quad (B_N^s)_{ij} = b(\zeta_j^s, \zeta_i^y), \ (B_{a,N}^s)_i = b(u_a, \zeta_i^y),$$
$$(Y_{d,N})_i = (y_d, \zeta_i^y)_D, \quad (D_N)_{ij} = (\zeta_i^y, \zeta_j^y)_D, \quad (U_N^s)_{ij} = (\zeta_i^s, \zeta_j^s)_U, \ (U_{a,N}^s)_i = (u_a, \zeta_i^s)_U,$$
$$(U_N^{\sigma_b,s})_{ij} = (\zeta_i^{\sigma_b}, \zeta_j^s)_U, \ (U_{b,N}^{\sigma_b})_i = (u_b, \zeta_i^{\sigma_b})_U, \ (U_{a,N}^{\sigma_b})_i = (u_a, \zeta_i^{\sigma_b})_U$$

and $1 \leq i,j \leq N_\bullet$ [see (3.3) and (3.14)].

Second, by solving (3.15) we have $\underline{s}_N \geq 0$ and through the definition of s we obtain a feasible—w.r.t. $u_a(\mu)$—approximation for the control by $u^s = s_N + u_a(\mu)$. In order to derive an error bound for $\|u - u^s\|_U$ we need, however, to analogously repeat the RB reduction for the second RB slack problem with $s_b = u_b(\mu) - u$. There we likewise introduce the RB space $S_{b,N}$, as well as its convex cone $S_{b,N}^+$ and follow the previous steps to obtain $s_{b,N} \geq 0$. Using this, we obtain a control approximation $u^{s_b} = u_b(\mu) - s_{b,N}$ that is feasible w.r.t. the constraint $u_b(\mu)$.

3.4.2 Primal-Slack Error Bound

In the following we will focus on the primal-slack error bound for $\|u - u^s\|_U$ w.r.t. $u_a(\mu)$. Similarly to the primal error bound in Theorem 1 we use residuals and properties of the bilinear and linear forms to derive a quadratic inequality in $\|u - u^s\|_U$. We consider the following RB primal-slack approximation $(y_N^s, p_N^s, u^s, \sigma_N, \sigma_{b,N}) \in Y_N \times Y_N \times U_{\text{ad}} \times \Sigma_N^+ \times \Sigma_{b,N}^+$, which depends on the solutions of the RB primal and slack problem. We define the corresponding errors $e_y^s = y - y_N^s$, $e_p^s = p - p_N^s$, $e_u^s = u - u^s$. Further, we revisit Definition 1 and insert on the r.h.s. of (3.5) the approximation $(y_N^s, p_N^s, u^s, \sigma_N, \sigma_{b,N})$ to obtain on the l.h.s. the residuals r_y^s, r_p^s, r_u^s.

We state the main result in the following theorem.

Theorem 2 (Primal-Slack Error Bound) *Let u, s_N, and $s_{b,N}$ be the optimal solutions of the FE primal problem* (**P**) *and the RB slack problems* ($\mathbf{S_N}$) *and its equivalent w.r.t. $u_b(\mu)$, respectively. Then the error in the optimal control satisfies* *for all parameters* $\mu \in \mathscr{D}$

$$\|e_u^s\|_U \leq \Delta_N^{\mathrm{pr-sl}}(\mu),$$

where $\Delta_N^{\mathrm{pr-sl}}(\mu) := c_1^s(\mu) + \sqrt{c_1^s(\mu)^2 + c_2^s(\mu)}$ *with nonnegative coefficients*

$$c_1^s(\mu) = \frac{1}{2\lambda}\left(\|r_u^s\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p^s\|_{Y'} + \lambda\|u^s - u^{s_b}\|_U\right), \tag{3.16a}$$

$$\begin{aligned} c_2^s(\mu) = \frac{1}{\lambda}\Bigg[&\frac{2}{\alpha_a^{\mathrm{LB}}}\|r_y^s\|_{Y'}\|r_p^s\|_{Y'} + \frac{1}{4}\left(\frac{C_D^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}(\|r_y^s\|_{Y'} + \gamma_b^{\mathrm{UB}}\|u^s - u^{s_b}\|_U)\right)^2 + (\sigma_N, s_N)_U \\ &+ \|u^s - u^{s_b}\|_U\left(\|r_u^s\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p^s\|_{Y'} + \sqrt{2(\sigma_N, \sigma_{b,N})_U}\right) + (\sigma_{b,N}, s_{b,N})_U\Bigg] \end{aligned} \tag{3.16b}$$

Proof Let the FE primal solution $(y, p, u, \sigma, \sigma_b)$ satisfy the optimality conditions (3.2). We follow the proof of Theorem 1, and derive analogously to (3.9) the inequality

$$\lambda\|e_u^s\|_U^2 + |e_y^s|_D^2 \leq \|r_y^s\|_{Y'}\|e_p^s\|_Y + \|r_p^s\|_{Y'}\|e_y^s\|_Y + \|r_u^s\|_{U'}\|e_u^s\|_U + (e_\sigma - e_{\sigma_b}, e_u^s)_U. \tag{3.17}$$

We first focus on $(e_\sigma, e_u^s)_U$ and exploit the feasibility of u^s w.r.t. $u_a(\mu)$. Again we have $(e_\sigma, e_u^s)_U = -(\sigma, u_a(\mu) - u)_U - (\sigma, s_N)_U + (\sigma_N, u_a(\mu) - u)_U + (\sigma_N, s_N)_U$, where the first three terms are non-positive and hence $(e_\sigma, e_u^s)_U \leq (\sigma_N, s_N)_U$. In order to bound $-(e_{\sigma_b}, e_u^s)_U$, we need to solve the second RB slack problem for u^{s_b} and derive

$$-(e_{\sigma_b}, e_u^s)_U = (e_{\sigma_b}, u^s - u^{s_b} + u^{s_b} - u)_U \leq \|e_{\sigma_b}\|_U\|u^s - u^{s_b}\|_U + (\sigma_{b,N}, s_{b,N})_U.$$

We restate that $\|e_{\sigma_b}\|_U \leq \|e_\sigma - e_{\sigma_b}\|_U + \sqrt{2(\sigma_N, \sigma_{b,N})_U}$ and $\|e_\sigma - e_{\sigma_b}\|_U$ is bounded by $\|e_\sigma - e_{\sigma_b}\|_U \leq \|r_u^s\|_{U'} + \lambda\|e_u^s\|_U + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\left(\|r_p^s\|_{Y'} + C_D^{\mathrm{UB}}|e_y^s|_D\right)$. Using the bounds for $(e_\sigma, e_u^s)_U - (e_{\sigma_b}, e_u^s)_U$ and the inequalities (3.8) in (3.17) we obtain

$$\begin{aligned} \lambda\|e_u^s\|_U^2 + |e_y^s|_D^2 \leq\ &\|e_u^s\|_U\left(\|r_u^s\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p^s\|_{Y'} + \lambda\|u^s - u^{s_b}\|_U\right) + (\sigma_N, s_N)_U \\ &+ \frac{2}{\alpha_a^{\mathrm{LB}}}\|r_y^s\|_{Y'}\|r_p^s\|_{Y'} + |e_y^s|_D\frac{C_D^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\left(\|r_y^s\|_{Y'} + \gamma_b^{\mathrm{UB}}\|u^s - u^{s_b}\|_U\right) \\ &+ \left(\|r_u^s\|_{U'} + \frac{\gamma_b^{\mathrm{UB}}}{\alpha_a^{\mathrm{LB}}}\|r_p^s\|_{Y'} + \sqrt{2(\sigma_N, \sigma_{b,N})_U}\right)\|u^s - u^{s_b}\|_U + (\sigma_{b,N}, s_{b,N})_U. \end{aligned}$$

It thus follows from employing Young's inequality to the $|e_y^s|_D$-term that $\|e_u^s\|_U^2 - 2c_1^s(\mu)\|e_u^s\|_U - c_2^s(\mu) \leq 0$, where $c_1^s(\mu)$ and $c_2^s(\mu)$ are given in (3.16). Solving the last inequality for the larger root yields $\|e_u^s\|_U \leq c_1^s(\mu) + \sqrt{c_1^s(\mu)^2 + c_2^s(\mu)} = \Delta_N^{\mathrm{pr-sl}}(\mu)$.

3.5 Greedy Sampling Procedure

The reduced basis spaces for the two-sided control-constrained optimal control problem in Sects. 3.3.1, and 3.4.1 are constructed using the greedy sampling procedure outlined in Algorithm 1. Suppose $\Xi_{\mathrm{train}} \subset \mathscr{D}$ is a finite but suitably large parameter train sample, $\mu^1 \in \Xi_{\mathrm{train}}$ is the initial parameter value, $N_{\max}$ the maximum number of greedy iterations, $\varepsilon_{\mathrm{tol,min}} > 0$ is a prescribed desired error tolerance, and $\Delta_N^{\bullet}(\mu)/\|u_N^{\bullet}(\mu)\|_U$, $\bullet \in \{\mathrm{pr}, \mathrm{pr{-}sl}\}$, is the primal or primal-slack error bound from (3.6) or (3.16) with $u_N^{\bullet} \in \{u_N, u^s\}$.

We make two remarks: First, by using the bounds $\Delta_N^{\bullet}(\mu)$, $\bullet \in \{\mathrm{pr}, \mathrm{pr{-}sl}\}$ we only refer to the bounds derived for the primal error $\|u - u_N\|_U$ and the slack error $\|u - u^s\|_U$ w.r.t. to $u_a(\mu)$. Therefore, using the primal-slack bound $\Delta_N^{\mathrm{pr-sl}}(\mu)$ in the greedy sampling procedure, we expect not only to construct an accurate RB space S_N for s_N but also an accurate RB space $S_{b,N}$ for $s_{b,N}$. Second, we comment on two special cases: (1) if one control constraint is fully active in each greedy step, i.e. we have, e.g., $u(\mu^n) = u_a,\ n = 1, \ldots, N$, we set $S_N = \{\}$ and $s_N = 0$ (analogously for u_b we set $S_{b,N} = \{\}$ and $s_{b,N} = 0$); and (2) if the control constraint is never active, i.e., for all snapshots $\sigma(\mu^n) = \sigma_b(\mu^n) = 0,\ n = 1, \ldots, N$, we set $\Sigma_N = \Sigma_N^+ = \Sigma_{b,N} = \Sigma_{b,N}^+ = \{\}$ and $\sigma_N = \sigma_{b,N} = 0$.

Algorithm 1 Greedy sampling procedure

```
Choose $\Xi_{\mathrm{train}} \subset \mathscr{D}$, $\mu^1 \in \Xi_{\mathrm{train}}$ (arbitrary), $N_{\max}$, and $\varepsilon_{\mathrm{tol,min}} > 0$
Set $N \leftarrow 1,\ Y_0 \leftarrow \{0\},\ U_0 \leftarrow \{0\},\ S_0 \leftarrow \{0\},\ S_{b,0} \leftarrow \{0\},\ \Sigma_0 \leftarrow \{0\},\ \Sigma_{b,0} \leftarrow \{0\}$
Set $\Delta_N^{\bullet}(\mu^N) \leftarrow \infty$
while $\Delta_N^{\bullet}(\mu)/\|u_N^{\bullet}(\mu^N)\|_U > \varepsilon_{\mathrm{tol,min}}$ and $N \leq N_{\max}$ do
    $Y_N \leftarrow Y_{N-1} \oplus \mathrm{span}\{\, y(\mu^N), p(\mu^N) \,\}$
    $U_N \leftarrow U_{N-1} \oplus \mathrm{span}\{\, u(\mu^N), \sigma(\mu^N), \sigma_b(\mu^N) \,\}$
    $S_N \leftarrow S_{N-1} \oplus \mathrm{span}\{\, s(\mu^N) \,\}$
    $S_{b,N} \leftarrow S_{b,N-1} \oplus \mathrm{span}\{\, s_b(\mu^N) \,\}$
    $\Sigma_N \leftarrow \Sigma_{N-1} \oplus \mathrm{span}\{\, \sigma(\mu^N) \,\}$
    $\Sigma_{b,N} \leftarrow \Sigma_{b,N-1} \oplus \mathrm{span}\{\, \sigma_b(\mu^N) \,\}$
    $\mu^{N+1} \leftarrow \arg\max_{\mu \in \Xi_{\mathrm{train}}} \ \Delta_N^{\bullet}(\mu)/\|u_N^{\bullet}(\mu^N)\|_U$
    $N \leftarrow N + 1$
end while
```

3.6 Numerical Results: Graetz Flow with Parametrized Geometry and Lower and Upper Control Constraints

We consider a Graetz flow problem, which describes a heat convection and conduction in a duct. The main goal of this example is to demonstrate the different properties of the approximations and their error bounds. The problem is parametrized by a varying Péclet number $\mu_1 \in [5, 18]$ and a geometry parameter $\mu_2 \in [0.8, 1.2]$. Hence, the parameter domain is $\mathscr{D} = [5, 18] \times [0.8, 1.2]$. The parametrized geometry is given by $\Omega(\mu) = [0, 1.5 + \mu_2] \times [0, 1]$ and is subdivided into three subdomains $\Omega_1(\mu) = [0.2\mu_2, 0.8\mu_2] \times [0.3, 0.7]$, $\Omega_2(\mu) = [\mu_2 + 0.2, \mu_2 + 1.5] \times [0.3, 0.7]$, and $\Omega_3(\mu) = \Omega(\mu) \setminus \{\Omega_1(\mu) \cup \Omega_2(\mu)\}$. A sketch of the domain is shown in Fig. 3.1. We impose boundary condition of homogeneous Neumann and of non-homogeneous Dirichlet type: $y_n = 0$ on $\Gamma_N(\mu)$, and $y = 1$ on $\Gamma_D(\mu)$. Thus the trial space is given by $Y(\mu) \subset Y_e(\mu) = \{v \in H^1(\Omega(\mu)); v|_{\Gamma_D(\mu)} = 1\}$. The amount of heat supply in the whole domain $\Omega(\mu)$ is regulated by the distributed control $u \in U(\mu) \subset U_e(\mu) = L^2(\Omega(\mu))$ and bounded by the lower and upper constraints $u_a = -0.5$ and $u_b = 1.25$. The observation domain is $\Omega_D(\mu) = \Omega_1(\mu) \cup \Omega_2(\mu)$ and the desired state is given by $y_d = 0.5$ on $\Omega_1(\mu)$ and $y_d = 2$ on $\Omega_2(\mu)$.

Overall, the parametrized optimal control problem is given by

$$\min_{\hat{y} \in Y(\mu), \hat{u} \in U(\mu)} J(\hat{y}, \hat{u}; \mu) = \frac{1}{2}|\hat{y} - y_d|^2_{D(\mu)} + \frac{\lambda}{2}\|\hat{u}\|^2_{L^2(\Omega(\mu))}$$

$$\text{s.t.} \quad \frac{1}{\mu_1}\int_{\Omega(\mu)} \nabla\hat{y} \cdot \nabla\phi \, \mathrm{d}x + \int_{\Omega(\mu)} \beta(x) \cdot \nabla\hat{y}\,\phi \, \mathrm{d}x = \int_{\Omega(\mu)} \hat{u}\,\phi \, \mathrm{d}x \quad \forall\phi \in Y(\mu),$$

$$(u_a, \rho)_{U(\mu)} \leq (\hat{u}, \rho)_{U(\mu)} \leq (u_b, \rho)_{U(\mu)} \quad \forall\rho \in U(\mu)^+,$$

for the given parabolic velocity field $\beta(x) = (x_2(1 - x_2), 0)^T$. The regularization parameter λ is fixed to 0.01.

After recasting the problem to a reference domain $\Omega = \Omega(\mu^{\text{ref}}) = [0, 2.5] \times [0, 1]$ for $\mu^{\text{ref}} = (5, 1)$, and introducing suitable lifting functions that take into account the non-homogeneous Dirichlet boundary conditions, we can reformulate

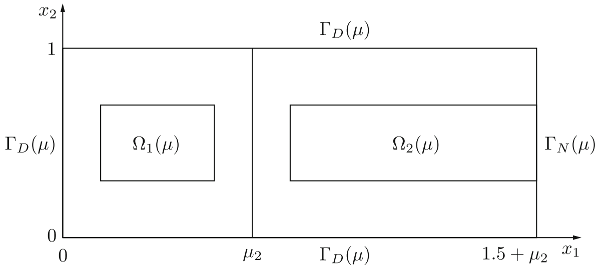


Fig. 3.1 Domain $\Omega(\mu)$ for the Graetz flow problem with distributed control

the problem in terms of the parameter-independent FE space $Y \subset Y_e = H_0^1(\Omega)$ and $U \subset U_e = L^2(\Omega)$ [12]. We then obtain the affine representation of all involved quantities with $Q_a = Q_f = 4$, $Q_b = Q_d = Q_u = Q_{yd} = 2$, and $Q_{ua} = 1$. The details of these calculations are very similar to the details presented by Rozza et al. [12] and Kärcher [5], and are thus omitted. The inner product for the state space is given by $(w, v)_Y = \frac{1}{\mu_1^{\text{ref}}} \int_\Omega \nabla w \cdot \nabla v \, dx + \frac{1}{2}(\int_\Omega \beta(x) \cdot \nabla w \, v \, dx + \int_\Omega \beta(x) \cdot \nabla v \, w \, dx)$ and we obtain a lower bound $\alpha_a^{\text{LB}}(\mu)$ for the coercivity constant by the so-called min-theta approach [12]. Note that for the control space we obtain a parameter-dependent inner product $(\cdot, \cdot)_{U(\mu)}$ from the affine geometry parametrization. Hence the control error is measured in the parameter-dependent energy norm $\|\cdot\|_{U(\mu)}$. The derivations of the primal and primal-slack error bounds remain the same in this case and they bound the control error in the energy norm.

Although the introduction of a domain parametrization seems to add an entirely new μ-dependence to the primal and the slack problems (**P**) and (**S**), the reductions and the error bound derivations can be analogously derived w.r.t. $(\cdot, \cdot)_{U(\mu)}$ instead of $(\cdot, \cdot)_{U(\mu^{\text{ref}})}$. Also the definition of the integrated space U_N in (3.3) remains, while in the inf-sup condition of Remark 2 we use $(\cdot, \cdot)_{U(\mu)}$ instead of $(\cdot, \cdot)_U$.

We choose a P_1 discretization for the state and adjoint, and a P_0 discretization for the control to obtain $\dim(Y) = \mathcal{N}_Y \approx 11{,}000$ and $\dim(U) = \mathcal{N}_U \approx 22{,}000$. The chosen discretization induces a discretization error of roughly 2%. In Fig. 3.2 we present control snapshots and associated active sets for two different parameters displayed on the reference domain $\Omega(\mu_2^{\text{ref}} = 1)$. We observe strongly varying control solutions and active sets.

We construct the RB spaces using the greedy procedure described in Algorithm 1 by employing an equidistant train sample $\Xi_{\text{train}} \subset \mathcal{D}$ of size $30 \cdot 30 = 900$ (log-scale in μ_1 and lin-scale in μ_2) and stop the greedy enrichment after 30 steps. We also introduce a test sample with $10 \cdot 5$ (log × lin) equidistant parameter points in $[5.2, 17.5] \times [0.82, 1.17] \subset \mathcal{D}$.

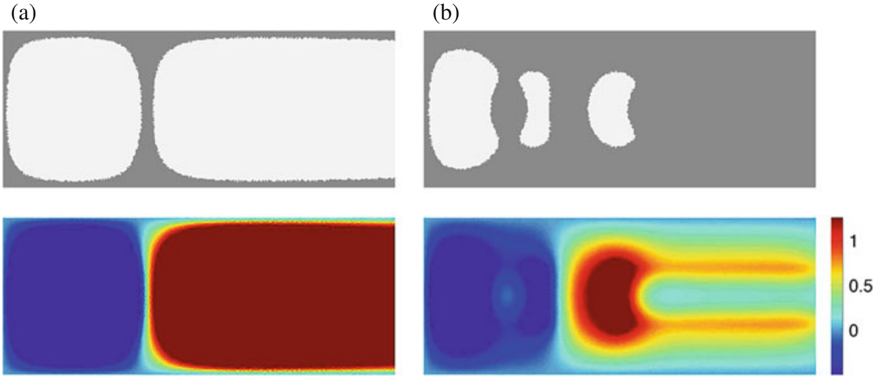


Fig. 3.2 Snapshots of active sets (*upper row*) and optimal control (*lower row*) on the reference domain. The active (inactive) sets are displayed in *light gray* (*gray*). (**a**) $\mu = (5, 0.8)$. (**b**) $\mu = (18, 1.2)$

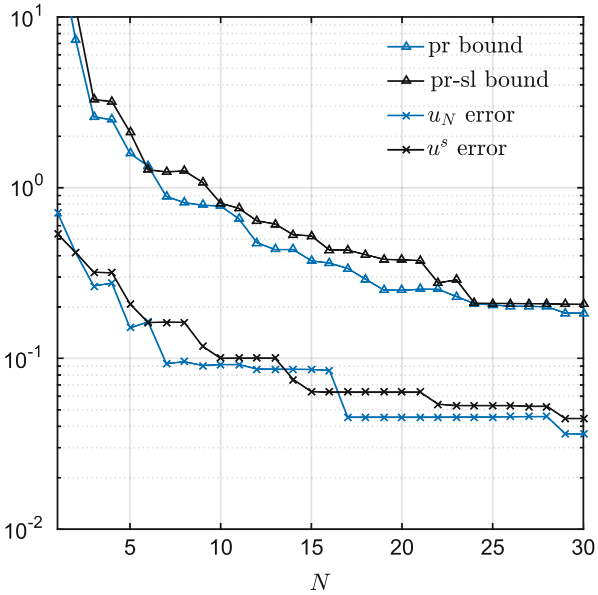


Fig. 3.3 Maximal relative control errors and bounds over the number of greedy iterations. For each N the maximal value over Ξ_{test} is displayed for both errors and both bound

In Fig. 3.3 we present, as a function of N, the resulting energy norm errors and bounds over Ξ_{test}. Here, the errors and bounds are defined as follows: the primal-slack bound is the maximum of $\Delta_N^{\text{pr-sl}}(\mu)/\|u(\mu)\|_{U(\mu)}$ over Ξ_{test}, the primal bound is the maximum of $\Delta_N^{\text{pr}}(\mu)/\|u(\mu)\|_{U(\mu)}$ over Ξ_{test}, and the u^s and u_N errors are the maxima of $\|u(\mu) - u^s(\mu)\|_{U(\mu)}/\|u(\mu)\|_{U(\mu)}$ and $\|u(\mu) - u_N(\mu)\|_{U(\mu)}/\|u(\mu)\|_{U(\mu)}$ over Ξ_{test}, respectively. We observe that both errors and both bounds decay very similarly. Quantitatively, the error bounds are comparable throughout all N, since the dominating primal-slack terms $\|r_u^s\|_{U'(\mu)}$ and $\lambda\|u^s - u^{s_b}\|_{U(\mu)}$ are comparable to the dominating primal terms $\|r_u\|_{U'(\mu)}$ and $\lambda(\delta_1+\delta_{1b})$, resulting in $\Delta_N^{\text{pr}}(\mu) \approx \Delta_N^{\text{pr-sl}}(\mu)$.

We briefly report the computational timings: the solution of the FE optimization problem takes $\approx$ 4 s (for a discretization error of 2%). The RB primal problem, for $N = 25$, is solved in $\approx$ 0.066 s and the RB slack problem is solved faster in $\approx$ 0.029 s, since $\dim(S_N) = 25$. We turn to the evaluation of error bounds: the primal bound takes 0.01 s, whereas the primal-slack bound, given σ_N, $\sigma_{b,N}$, takes 0.0065 s. From this we can conclude that for $N = 25$ the overall cost for one primal bound evaluation is roughly $0.076\,\text{s} = 0.066\,\text{s} + 0.01\,\text{s}$ and for the primal-slack bound evaluation is roughly $0.13 \approx 0.029 + 0.029 + 0.066 + 0.0065\,\text{s}$, since it relies on three RB solutions.

3.7 Conclusions

In this paper we extended the ideas from [1] to propose two certified reduced basis approaches for distributed elliptic optimal control problems with two-sided control constraints: a primal and a primal-slack approach. Albeit the reduction for the primal approach was straightforward, the primal-slack approach needed more consideration. We proposed for each constraint a corresponding RB slack problem with an additional Lagrange multiplier. The primal a posteriori error bound from [1] could be extended for the two-sided case by special properties of the Lagrange multipliers of the two-sided problem. The primal-slack error bound also relies on these properties and in addition uses three RB solutions to derive an $\mathscr{N}$-independent error bound. Both the primal and slack RB approximation can be evaluated efficiently using the standard offline-online decomposition. However, on the one hand the primal error bound depends on the FE control dimension and on the other hand the primal-slack error bound relies on three reduced order optimization problems.

References

1. Bader, E., Kärcher, M., Grepl, M.A., Veroy, K.: Certified reduced basis methods for parametrized distributed optimal control problems with control constraints. SIAM J. Sci. Comput. **38**(6), A3921–A3946 (2016)
2. Balajewicz, M., David, A., Farhat, C.: Projection-based model reduction for contact problems. Int. J. Numer. Methods Eng. **106**(8), 644–663 (2016)
3. Haasdonk, B., Salomon, J., Wohlmuth, B.: A reduced basis method for parametrized variational inequalities. SIAM J. Numer. Anal. **50**(5), 2656–2676 (2012)
4. Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications, vol. 23. Springer, Berlin (2009)
5. Kärcher, M.: Certified reduced basis methods for parametrized PDE-constrained optimization problems. PhD thesis, RWTH Aachen University (2016)
6. Kärcher, M., Grepl, M.A.: A certified reduced basis method for parametrized elliptic optimal control problems. ESAIM Control Optim. Calculus Var. **20**(2), 416–441 (2013)
7. Kärcher, M., Grepl, M.A., Veroy, K.: Certified reduced basis methods for parameterized distributed optimal control problems. Technical report (2014)
8. Lions, J.L.: Optimal Control of Systems Governed by Partial Differential Equations. Springer, New York (1971)
9. Maday, Y., Patera, A.T., Rovas, D.V.: A blackbox reduced-basis output bound method for noncoercive linear problems. In: Cioranescu, D., Lions, J.L. (eds.) Studies in Mathematics and Its Applications, vol. 31, pp. 533–569. Elsevier Science B.V., Amsterdam (2002)
10. Negri, F., Rozza, G., Manzoni, A., Quarteroni, A.: Reduced basis method for parametrized elliptic optimal control problems. SIAM J. Sci. Comput. **35**(5), A2316–A2340 (2013)
11. Negri, F., Manzoni, A., Rozza, G.: Reduced basis approximation of parametrized optimal flow control problems for the stokes equations. Comput. Math. Appl. **69**(4), 319–336 (2015)
12. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Arch. Comput. Methods Eng. **15**(3), 229–275 (2008)
13. Tröltzsch, F., Volkwein, S.: POD a-posteriori error estimates for linear-quadratic optimal control problems. Comput. Optim. Appl. **44**, 83–115 (2009)
14. Zhang, Z., Bader, E., Veroy, K.: A slack approach to reduced-basis approximation and error estimation for variational inequalities. C.R. Math. **354**(3), 283–289 (2016)

Chapter 4
A Reduced Basis Method with an Exact Solution Certificate and Spatio-Parameter Adaptivity: Application to Linear Elasticity

Masayuki Yano

Abstract We present a reduced basis method for parametrized linear elasticity equations with two objectives: providing an error bound with respect to the exact weak solution of the PDE, as opposed to the typical finite-element "truth", in the online stage; providing automatic adaptivity in both physical and parameter spaces in the offline stage. Our error bound builds on two ingredients: a minimum-residual mixed formulation with a built-in bound for the dual norm of the residual with respect to an infinite-dimensional function space; a combination of a minimum eigenvalue bound technique and the successive constraint method which provides a lower bound of the stability constant with respect to the infinite-dimensional function space. The automatic adaptivity combines spatial mesh adaptation and greedy parameter sampling for reduced bases and successive constraint method to yield a reliable online system in an efficient manner. We demonstrate the effectiveness of the approach for a parametrized linear elasticity problem with geometry transformations and parameter-dependent singularities induced by cracks.

4.1 Introduction

Reduced basis (RB) methods provide rapid and reliable solution of parametrized partial differential equations (PDEs), including linear elasticity equations, in real-time and many-query applications; see, e.g., a review paper [13] and early applications to linear elasticity in [4, 7, 10, 14]. However, until recently, RB methods have focused on approximating the high-fidelity "truth" solution—typically a finite element (FE) solution on a prescribed mesh—and not the exact solution of the PDE, which is of actual interest. Classical RB methods *assume* that the "truth" model is sufficiently accurate to serve as an surrogate for the exact PDE. However, in practice, satisfying the assumption requires a careful mesh construction especially

M. Yano (✉)
University of Toronto Institute for Aerospace Studies, 4925 Dufferin St, Toronto, ON, Canada M3H 5T6
e-mail: myano@utias.utoronto.ca

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_4

in the presence of sharp corners and cracks (as done in [4]), and in any event the assumption is never rigorously verified for all parameter values. In this work, we present a RB method which provides a certificate with respect to the *exact* solution of the parametrized PDE and automatically produces a reduced model that meets the desired tolerance through automatic adaptivity, eliminating the issue of the "truth".

Specifically, we present a RB method for linear elasticity problems that provides

1. error bounds with respect to the exact solution in energy norm or for functional outputs for any parameter value in the online stage;
2. automatic adaptivity in physical space and parameter space to control the error with respect to the exact solution;
3. a strict offline-online computational decomposition such that the online computational cost is independent of the offline FE solves.

Item 3 provides rapidness, as in the case for the standard RB method. Items 1 and 2, which provide certification and adaptivity with respect to the exact solution, distinguish our method from the standard RB method.

Recently, a number of RB methods has been proposed to provide error bounds with respect to the exact solution. Ali et al. [1] consider a RB method based on snapshots generated by an adaptive wavelet method. Ohlberger and Schindler [8] considers a RB method for multiscale problems with an error bound with respect to the exact solution. We have also introduced RB methods which provide error bounds with respect to the exact solution using the complementary variational principle [15] and using a minimum-residual mixed formulation [16, 17]. This work shares a common goal with the above recent works in the RB community.

The error certification and adaptation approach that we present in this paper is an extension of the method we introduced in [17] for scalar equations to linear elasticity equations with piecewise-affine geometry transformations. We provide a solution approximation and an upper bound of the residual dual norm using a minimum-residual mixed formulation. We provide a lower bound of the stability constant using a version of the successive constraint method (SCM) [5], which has been extended to provide bounds relative to an appropriate infinite-dimensional function space by appealing to Weinstein's method and a residual-based bounding technique. In extending the approach to linear elasticity, special attention is paid to the treatment of rigid-body rotation modes and the construction of the dual space in the presence of geometry transformations.

The paper is organized as follows. Section 4.2 defines the problem of interest. Section 4.3 presents our residual bound procedure. Section 4.4 presents our stability-constant bound procedure. Section 4.5 presents the error bound. Section 4.6 presents spatio-parameter adaptive algorithms. Section 4.7 presents numerical results.

4.2 Preliminaries

4.2.1 Problem Statement

Notations In order to describe tensor operations that appear in linear elasticity, we now fix the notations. Given a order-2 tensor $\underline{w}$, we "reshape" it as a vector $w \in \mathbb{R}^{d^2}$ with entries $(w)_{i \cdot d+j} = \underline{w}_{ij}$. Similarly, given a order-4 tensor $\underline{A}$, we "reshape" it as a matrix $A \in \mathbb{R}^{d^2 \times d^2}$ with entries $(A)_{i \cdot d+j, k \cdot d+l} = \underline{A}_{ijkl}$. These reshaped notations allow us to precisely express operations on order-2 and -4 tensors using the standard linear algebra notations without introducing explicit indices.

Using the convection, the derivative of a vector field $v : \Omega \to \mathbb{R}^d$ evaluated at x is expressed as a vector $v(x) \in \mathbb{R}^{d^2}$ with entries $(\nabla v(x))_{i \cdot d+j} = \frac{\partial v_i}{\partial x_j}$. Similarly, the divergence of a order-2 tensor field $q : \Omega \to \mathbb{R}^{d^2}$ evaluated at x is expressed as a vector $\nabla q(x) \in \mathbb{R}^d$ with entries $(\nabla q(x))_i = \sum_{j=1}^d \frac{\partial q_{ij}}{\partial x_j}$; the evaluation of q at x in the direction of $n \in \mathbb{R}^d$ is expressed as a vector $n \cdot q(x) \in \mathbb{R}^d$ with entries $(n \cdot q(x))_i = \sum_{j=1}^d q_{ij}(x) n_j$.

Problem Description over a Parametrized Domain We first introduce a P-dimensional parameter domain $\mathcal{D} \subset \mathbb{R}^P$. We next introduce a d-dimensional parametrized physical domain $\tilde{\Omega}(\mu) \subset \mathbb{R}^d$ with a Lipchitz boundary $\partial \tilde{\Omega}(\mu)$. For each component $i = 1, \dots, d$, the boundary $\partial \tilde{\Omega}(\mu)$ is decomposed into a Dirichlet part $\tilde{\Gamma}_{D,i}(\mu)$ and a Neumann part $\tilde{\Gamma}_{N,i}(\mu)$ such that $\overline{\partial \tilde{\Omega}(\mu)} = \overline{\tilde{\Gamma}_{D,i}(\mu)} \cup \overline{\tilde{\Gamma}_{N,i}(\mu)}$. We then introduce a Sobolev space $\mathcal{V}(\tilde{\Omega}) = \{\tilde{v} \in (H^1(\tilde{\Omega}))^d \mid \tilde{v}_i|_{\tilde{\Gamma}_{D,i}} = 0,\ i = 1, \dots, d\}$, where $H^1(\tilde{\Omega})$ is the standard H^1 Sobolev space over $\tilde{\Omega}$. (See, e.g., [2].)

We now introduce order-4 tensors, unwrapped as $d^2 \times d^2$ matrices, associated with our linear elasticity problem. We first introduce the strain tensor operator $E \in \mathbb{R}^{d^2 \times d^2}$ such that $E \nabla \tilde{v}(\tilde{x}) \in \mathbb{R}^{d^2}$ is the reshaped strain tensor. We next introduce a parametrized stiffness tensor field $\tilde{K} : \mathcal{D} \times \tilde{\Omega} \to \mathbb{R}^{d^2 \times d^2}$; by definition the stiffness tensor is symmetric positive definite for all $\mu \in \mathcal{D}$ and $\tilde{x} \in \tilde{\Omega}$. We also introduce the associated parametrized compliance tensor field $\tilde{C} : \mathcal{D} \times \tilde{\Omega} \to \mathbb{R}^{d^2 \times d^2}$. The stiffness and compliance tensor are related by $\tilde{K}(\mu; \tilde{x}) \tilde{C}(\mu; \tilde{x}) = I_{d^2}$, where I_{d^2} denotes the $d^2 \times d^2$ identity matrix.

We now consider the following weak formulation of linear elasticity: given $\mu \in \mathcal{D}$, find $\tilde{u}(\mu) \in \mathcal{V}(\tilde{\Omega}(\mu))$ such that

$$a_{\tilde{\Omega}(\mu)}(\tilde{u}(\mu), \tilde{v}; \mu) = \ell_{\tilde{\Omega}(\mu)}(\tilde{v}; \mu) \quad \forall \tilde{v} \in \mathcal{V}(\tilde{\Omega}) \tag{4.1}$$

where

$$a_{\tilde{\Omega}(\mu)}(\tilde{w}, \tilde{v}; \mu) = \int_{\tilde{\Omega}(\mu)} \tilde{\nabla} \tilde{v}^T E^T \tilde{K}(\mu) E \tilde{\nabla} \tilde{w} d\tilde{x}, \tag{4.2}$$

$$\ell_{\tilde{\Omega}(\mu)}(\tilde{v}; \mu) = \int_{\tilde{\Omega}(\mu)} \tilde{v}^T \tilde{f}(\mu) d\tilde{x} + \int_{\tilde{\Gamma}_N(\mu)} \tilde{v}^T \tilde{g}(\mu) d\tilde{s}.$$

Here, $\tilde{f}(\mu)$ is the body force on the solid, $\tilde{g}(\mu)$ is the traction force on the Neumann boundaries, and the subscript $\tilde{\Omega}(\mu)$ on the forms emphasizes the problem is defined over a parameterized physical domain.

Reference-Domain Formulation Following the standard approach to treat parametrized geometric variations in the RB method (see, e.g., [13, 14]), we recast the problem over the parametrized domain $\tilde{\Omega}(\mu)$ to a parameter-independent reference domain Ω. Specifically, we consider each point $\tilde{x} \in \tilde{\Omega}(\mu)$ to be associated with a unique point $x \in \Omega$ by a piecewise affine map. We denote the Jacobian of the parametrized map by $J(\mu) \in \mathbb{R}^{d\times d}$ and the associated determinant by $|J(\mu)|$. Similarly, we denote the Jacobian associated with the mapping of a boundary segment by $|\partial J(\mu)|$. We also introduce a block matrix $Y = I_d \otimes J(\mu) \in \mathbb{R}^{d^2\times d^2}$ that facilitates transformation of tensors; here $\otimes$ is the Kronecker product.

We now introduce a Sobolev space over Ω,

$$\mathcal{V} \equiv \mathcal{V}(\Omega) \equiv \{v \in (H^1(\Omega))^d \mid v_i|_{\Gamma_{D_i}} = 0,\ i = 1,\dots,d\}$$

endowed with an inner product

$$(w,v)_{\mathcal{V}} \equiv \int_\Omega \nabla v^T \nabla w dx + \int_\Omega v^T w dx + \int_{\Gamma_N} v^T w ds \tag{4.3}$$

and the associated induced norm $\|v\|_{\mathcal{V}} \equiv \sqrt{(v,v)_{\mathcal{V}}}$. We then introduce a weak formulation that is equivalent to (4.1) but is associated with the reference domain: given $\mu \in \mathcal{D}$, find $u(\mu) \in \mathcal{V}$ such that

$$a(u(\mu), v; \mu) = \ell(v;\mu) \quad \forall v \in \mathcal{V}, \tag{4.4}$$

where

$$a(w,v;\mu) = \int_\Omega \nabla v^T Y(\mu)^{-1} E K(\mu) E Y(\mu)^{-T} \nabla w |J(\mu)| dx$$
$$\ell(v;\mu) = \int_\Omega v^T f(\mu) |J(\mu)| dx + \int_{\Gamma_N} v^T g(\mu) |\partial J(\mu)| ds.$$

Here the tensor fields in the physical and reference domains are related by $\tilde{v}(\tilde{x}) = v(x)$, $\tilde{K}(\mu;\tilde{x}) = K(\mu;x)$, $\tilde{f}(\mu;\tilde{x}) = f(\mu;x)$, and $\tilde{g}(\mu;\tilde{x}) = g(\mu;x)$. We readily verify that $a(\cdot,\cdot;\mu)$ is symmetric and bounded in $\mathcal{V}$. We also note that $a(\cdot,\cdot;\mu)$ is coercive in $\mathcal{V}$ due to the Korn inequality and the trace theorem [2]; we denote the associated energy norm by $|||\cdot|||_\mu \equiv \sqrt{a(\cdot,\cdot;\mu)}$.

Remark 1 In the standard RB formulation [13], we simply treat the elasticity equation as a vector-valued equation with the stiffness matrix $\hat{K}(\mu) \equiv |J(\mu)| Y(\mu)^{-1} E K(\mu) E Y(\mu)^{-T}$. Unfortunately, our exact error-bound formulation does not permit this simple treatment; our formulation [17] requires the inverse of the stiffness matrix, while the matrix $\hat{K}(\mu)$ is singular because $EK(\mu)E$ is rank-deficient. We will keep the explicit representation of the stiffness matrix to clearly show how our bound formulation for linear elasticity circumvents the issue.

Assumptions We clarify the set of assumptions for our RB formulation. First, we assume that the stiffness tensor $K(\mu)$, the compliance tensor $C(\mu)$, the body force $f(\mu)$, and the boundary traction force $g(\mu)$ each admit a decomposition that is affine in functions of parameter: $K(\mu) = \sum_{q=1}^{Q_K} \Theta_q^K(\mu) K_q$, $C(\mu) = \sum_{q=1}^{Q_C} \Theta_q^C(\mu) C_q$, $f(\mu) = \sum_{q=1}^{Q_f} \Theta_q^f(\mu) f_q$, and $g(\mu) = \sum_{q=1}^{Q_g} \Theta_q^g(\mu) g_q$, where $K_q : \Omega \to \mathbb{R}^{d^2 \times d^2}$, $C_q : \Omega \to \mathbb{R}^{d^2 \times d^2}$, $f_q : \Omega \to \mathbb{R}^d$, and $g_q : \Omega \to \mathbb{R}^d$ are parameter-independent fields, and $\Theta_q^K : \mathcal{D} \to \mathbb{R}$, $\Theta_q^C : \mathcal{D} \to \mathbb{R}$, $\Theta_q^f : \mathcal{D} \to \mathbb{R}$, and $\Theta_q^g : \mathcal{D} \to \mathbb{R}$ are parameter-dependent functions. Second, we assume that the mapping from the reference domain Ω to the physical domain $\tilde{\Omega}(\mu)$ is piecewise affine such that both the Jacobian $J(\mu)$ and the inverse Jacobian $J(\mu)^{-1}$ admit a decomposition that are affine in functions of parameter: $J(\mu) = \sum_{q=1}^{Q_J} \Theta_q^J(\mu) J_q$ and $J(\mu)^{-1} = \sum_{q=1}^{Q_{J^{\rm inv}}} \Theta_q^{J^{\rm inv}}(\mu) J_q^{\rm inv}$. Finally, we assume that the fields $K(\mu)$, $C(\mu)$, $f(\mu)$, and $g(\mu)$ are piecewise polynomials such that we can integrate the fields exactly using standard quadrature rules.

4.2.2 *Abstract Error Bounds: Energy Norm and Compliance Output*

To simplify the presentation of our formulation, we introduce a parametrized inner product

$$(w, v)_{\mathcal{W}(\mu;\delta)} = a(w, v) + \delta (w, v)_{\mathcal{V}}$$

and the associated induced norm $\|w\|_{\mathcal{W}(\mu;\delta)} \equiv \sqrt{(w, w)_{\mathcal{W}(\mu;\delta)}}$ for a parameter $\mu \in \mathcal{D}$ and a weight $\delta \in \mathbb{R}_{>0}$. Here $a(\cdot, \cdot; \mu)$ is the bilinear form (4.2), and $(\cdot, \cdot)_{\mathcal{V}}$ is the inner product (4.3). The parametrized norm is related to the energy norm by $\|v\|^2_{\mathcal{W}(\mu;\delta)} = |||v|||^2_\mu + \delta \|v\|^2_{\mathcal{V}}$. For any $\delta \in \mathbb{R}_{>0}$, the norm $\| \cdot \|_{\mathcal{W}(\mu;\delta)}$ is equivalent to the energy norm $||| \cdot |||_\mu$, which in turn is equivalent to $\| \cdot \|_{H^1(\Omega)}$. The role of δ in our formulation is discussed in Sect. 4.5.

In order to bound the error, we now introduce the residual form

$$r(v; w; \mu) \equiv \ell(v; \mu) - a(w, v; \mu) \quad \forall w, v \in \mathcal{V} \tag{4.5}$$

and the associated dual norm $\|r(\cdot; w; \mu)\|_{\mathcal{W}'(\mu;\delta)} \equiv \sup_{v \in \mathcal{V}} \frac{r(v;w;\mu)}{\|v\|_{\mathcal{W}(\mu;\delta)}}$. We also introduce the stability constant

$$\alpha(\mu; \delta) \equiv \inf_{v \in \mathcal{V}} \frac{|||v|||^2_\mu}{\|v\|^2_{\mathcal{W}(\mu;\delta)}}. \tag{4.6}$$

The following proposition bounds the energy norm of the error.

Proposition 2 *Given $\mu \in \mathcal{D}$ and an approximation $w \in \mathcal{V}$, the error is bounded by*

$$|||u(\mu) - w|||_{\mu} \leq \frac{1}{(\alpha(\mu;\delta))^{1/2}} \|r(\cdot; w; \mu)\|_{\mathcal{W}'(\mu;\delta)},$$

where $r(\cdot,\cdot;\cdot)$ is the residual form (4.5)*, and $\alpha(\cdot,\cdot)$ is the stability constant* (4.6)*.*

Proof See, e.g., Rozza et al. [13].

We can also construct an error bound for the compliance output $s(\mu) \equiv \ell(u(\mu);\mu)$.

Proposition 3 *Let the compliance output associated with an approximation $w \in \mathcal{V}$ be $\hat{s}(\mu) \equiv \ell(w;\mu) + r(w;w;\mu)$, where $r(\cdot;\cdot;\cdot)$ is the residual form* (4.5)*. Then, the error in the compliance output is bounded by*

$$|s(\mu) - \hat{s}(\mu)| \leq \frac{1}{\alpha(\mu;\delta)} \|r(\cdot; w; \mu)\|^2_{\mathcal{W}'(\mu;\delta)}.$$

Proof We suppress μ for brevity. It follows $s(\mu) - \hat{s}(\mu) = \ell(u) - (\ell(w) + r(w;w)) = \ell(u) - \ell(w) - \ell(w) + a(w,w) = \ell(u-w) - a(u-w,w) = a(u-w,u-w) = |||u-w|||^2_{\mu}$. Proposition 2 then yields the desired result.

The energy-norm and compliance-output error bound both require the same ingredients: an upper bound of the dual norm of the residual and a lower bound of the stability constant. In the next two sections, we develop offline-online efficient computational procedures for both of these quantities.

Remark 4 The output bound framework may be extended to any linear functional output by introducing the adjoint equation; see, e.g., Rozza et al. [13].

4.3 Upper Bound of the Dual Norm of the Residual

4.3.1 Bound Form

Our bound formulation is based on a mixed formulation and requires a dual field [16, 17]. Our dual space over a physical domain is the $H(\mathrm{div})$-conforming space

$$\mathcal{Q}(\tilde{\Omega}(\mu)) \equiv \{\tilde{q} \in (L^2(\tilde{\Omega}(\mu)))^{d^2} \mid \tilde{\nabla} \cdot \tilde{q} \in (L^2(\tilde{\Omega}(\mu)))^{d}\}.$$

The dual space over the reference domain is given by

$$\mathcal{Q} \equiv \{q \in (L^2(\Omega))^{d^2} \mid \nabla \cdot q \in (L^2(\Omega))^{d}\}.$$

We relate a field in a physical domain $\tilde{q} \in \mathcal{Q}(\tilde{\Omega}(\mu))$ and a field in the reference domain $q \in \mathcal{Q}$ by the Piola transformation, $\tilde{q}(\tilde{x}) = |J(\mu)|^{-1} Yq(x)$. The Piola transformation has an important property that it preserves $H(\text{div})$-conformity.

The following proposition introduces a version of the bound form introduced in [17] extended to linear elasticity equations with geometry transformations.

Proposition 5 *For any $w \in \mathcal{V}$, $q \in \mathcal{Q}$, $\mu \in \mathcal{D}$, and $\delta \in \mathbb{R}_{>0}$,*

$$\|r(\cdot; w; \mu)\|_{\mathcal{W}'(\mu;\delta)} \leq (F(w, q; \mu; \delta))^{1/2},$$

where the bound form is given by

$$\begin{aligned} F(w, q; \mu; \delta) = {} & \| |J(\mu)|^{-1/2} C(\mu)^{1/2} Y(\mu) q - |J(\mu)|^{1/2} K(\mu)^{1/2} E Y(\mu)^{-T} \nabla w \|^2_{L^2(\Omega)} \\ & + \delta^{-1} \|Y(\mu)^{-1}(I - E) Y(\mu) q\|^2_{L^2(\Omega)} + \delta^{-1} \|\nabla \cdot q + f(\mu)|J(\mu)|\|^2_{L^2(\Omega)} \\ & + \delta^{-1} \|g(\mu)|\partial J(\mu)| - n \cdot q\|^2_{L^2(\Gamma_N)} \end{aligned} \tag{4.7}$$

Proof For notational simplicity, we suppress μ from parameter-dependent operators and forms in the proof. For all $v \in \mathcal{V}$, $w \in \mathcal{V}$, $q \in \mathcal{Q}$, and $\delta \in \mathbb{R}_{>0}$,

$$\begin{aligned} & r(v; w; \mu; \delta) \\ & = \int_\Omega v^T f |J| dx + \int_{\Gamma_N} v^T g |\partial J| ds - \int_\Omega \nabla v^T Y^{-1} E^T K E Y^{-T} \nabla w |J| dx \\ & \quad + \int_\Omega v^T \nabla \cdot q dx + \int_\Omega \nabla v^T q dx - \int_{\Gamma_N} v^T n \cdot q ds \\ & = \int_\Omega \nabla v^T Y^{-1} E^T K |J| (|J|^{-1} C Y q - E Y^{-T} \nabla w) dx + \int_\Omega \nabla v^T Y^{-1} (I - E) Y q dx \\ & \quad + \int_\Omega v^T (\nabla \cdot q + f|J|) dx + \int_{\Gamma_N} v^T (g |\partial J| - n \cdot q) ds \\ & \leq (\| |J|^{1/2} K^{1/2} E Y^{-1} \nabla v \|^2_{L^2(\Omega)} + \delta \|\nabla v\|^2_{L^2(\Omega)} + \delta \|v\|^2_{L^2(\Omega)} + \delta \|v\|^2_{L^2(\Gamma_N)})^{1/2} \\ & \quad (\| |J|^{-1/2} C^{1/2} Y q - |J|^{1/2} K^{1/2} E Y^{-T} \nabla w \|^2_{L^2(\Omega)} + \delta^{-1} \|Y^{-1}(I - E) Y q\|^2_{L^2(\Omega)} \\ & \qquad + \delta^{-1} \|\nabla \cdot q + f|J|\|^2_{L^2(\Omega)} + \delta^{-1} \|g|\partial J| - n \cdot q\|^2_{L^2(\Gamma_N)})^{1/2} \\ & = \|v\|_{\mathcal{W}(\mu;\delta)} (F(w, q; \mu; \delta))^{1/2}. \end{aligned}$$

Note, the second line of the first equality vanishes by the Green's theorem. Hence, $\|r(\cdot; w; \mu; \delta)\|_{\mathcal{W}'(\mu;\delta)} = \sup_{v \in \mathcal{V}} r(v; w; \mu; \delta)/\|v\|_{\mathcal{W}(\mu;\delta)} \leq (F(w, q; \mu; \delta))^{1/2}$, $\forall q \in \mathcal{Q}$, which is the desired inequality.

The bound form (4.7) for linear elasticity is similar to the bound form for scalar equations introduced in [17]. However, the bound form differs in that it includes the “asymmetric penalty” term $\|Y(\mu)^{-1}(I - E)Y(\mu)q\|^2_{L^2(\Omega)}$; this term penalizes asymmetry in the dual tensor field in the *physical domain*, $\tilde{q} \in \tilde{Q}(\mu)$. In our bounding procedure, this term arises because the linear elasticity equation has zero energy with respect to not only translation but also rotation. In fact, the presence of this term is closely related to the complementary variational principle for elasticity equations requiring a symmetric dual field [9], as discussed in detail in Sect. 4.5.

The form (4.7) admits a decomposition into a quadratic, linear, and constant forms:

$$F(w,p;\mu;\delta) = G((w,p),(w,p);\mu;\delta) - 2L((w,p);\mu;\delta) + H(\mu;\delta).$$

We here omit the explicit expressions for brevity and refer to a similar decomposition *without the “asymmetric penalty” term* in [17]. The forms G, L, and H inherit the affine decomposition of the parametrized operators $K(\mu)$, $C(\mu)$, $f(\mu)$, $g(\mu)$, $J(\mu)$ and $J(\mu)^{-1}$, which makes the bound form F amenable to offline-online computational decomposition. In addition, the form $G(\cdot,\cdot;\mu;\delta)$ is coercive and bounded in $\mathcal{V} \times \mathcal{Q}$; the proof relies on Korn’s inequality and is omitted here for brevity.

4.3.2 Minimum-Bound Solutions and Approximations

Exact Solution We consider the following minimum bound problem: given $\mu \in \mathcal{D}$ and $\delta \in \mathbb{R}_{>0}$, find $(u(\mu), p(\mu)) \in \mathcal{V} \times \mathcal{Q}$ such that

$$(u(\mu), p(\mu)) = \underset{w\in\mathcal{V},\ q\in\mathcal{Q}}{\arg\inf}\ F(w,q;\mu;\delta).$$

The associated Euler-Lagrange equation is the following: given $\mu \in \mathcal{D}$, find $(u(\mu), p(\mu)) \in \mathcal{V} \times \mathcal{Q}$ such that

$$G((u(\mu),p(\mu)),(v,q);\mu;\delta) = L((v,q);\mu;\delta) \quad \forall v \in \mathcal{V},\ \forall q \in \mathcal{Q}.$$

The problem is wellposed due to the coercivity and boundedness of G in $\mathcal{V} \times \mathcal{Q}$.

We can readily show that the primal solution $u(\mu)$ is the weak solution of the original problem (4.4), and the dual solution $p(\mu)$ in the reference domain is related to the primal solution by $|J(\mu)|^{-1}Y(\mu)p(\mu) = K(\mu)EY^{-T}(\mu)\nabla u(\mu)$. The associated residual bound is 0 as expected. Equivalently, the dual solution and the primal solution are related in the physical domain by $\tilde{p}(\mu) = \tilde{K}(\mu)E\tilde{\nabla}\tilde{u}(\mu)$; the dual solution in the physical domain is the stress field. The tensor associated with the dual field $\tilde{p}(\mu)$ is symmetric in the physical domain, which is consistent with the constitutive relation, but is not symmetric in the reference domain.

FE For a FE approximation of the minimum bound problem, we first introduce a primal FE space $\mathcal{V}^{\mathcal{N}}$ of H^1-conforming Lagrange elements and a dual FE space $\mathcal{Q}^{\mathcal{N}}$ of $H(\mathrm{div})$-conforming Raviart-Thomas elements [11]. We then consider the minimum-bound FE approximation: given $\mu \in \mathcal{D}$ and $\delta \in \mathbb{R}_{>0}$, find $(u^{\mathcal{N}}(\mu), p^{\mathcal{N}}(\mu)) \in \mathcal{V}^{\mathcal{N}} \times \mathcal{Q}^{\mathcal{N}}$ such that

$$G((u^{\mathcal{N}}(\mu), p^{\mathcal{N}}(\mu)), (v, q); \mu; \delta) = L((v, q); \mu; \delta) \quad \forall v \in \mathcal{V}^{\mathcal{N}},\ \forall q \in \mathcal{Q}^{\mathcal{N}}. \tag{4.8}$$

The problem is wellposed due to the coercivity and boundedness of G and L. The dual norm of the residual is bounded by $\|r(\cdot; u^{\mathcal{N}}(\mu); \mu)\|_{\mathcal{W}'(\mu;\delta)} \leq F(u^{\mathcal{N}}(\mu), p^{\mathcal{N}}(\mu); \mu; \delta)^{1/2}$.

RB For a RB approximation of the minimum bound problem, we first introduce primal and dual RB spaces $\mathcal{V}_N = \mathrm{span}\{\xi_i\}_{i=1}^N \subset \mathcal{V}$ and $\mathcal{Q}_N = \mathrm{span}\{\eta_i\}_{i=1}^N \subset \mathcal{Q}$. We then introduce a minimum-bound RB approximation: given $\mu \in \mathcal{D}$ and $\delta \in \mathbb{R}_{>0}$, find $(u_N(\mu), p_N(\mu)) \in \mathcal{V}_N \times \mathcal{Q}_N$ such that

$$G((u_N(\mu), p_N(\mu)), (v, q); \mu; \delta) = L((v, q); \mu; \delta) \quad \forall v \in \mathcal{V}_N,\ \forall q \in \mathcal{Q}_N.$$

The problem is again wellposed due to the coercivity and boundedness of G and L. The dual norm of the residual is bounded by $\|r(\cdot; u_N(\mu); \mu)\|_{\mathcal{W}'(\mu;\delta)} \leq F(u_N(\mu), p_N(\mu); \mu; \delta)^{1/2}$.

4.4 Stability Constant

4.4.1 Transformation of the Stability Constant

We recall that a lower bound of the stability constant $\alpha(\mu; \delta)$ is needed to bound the energy norm of the error. In our approach, we do not compute a lower bound of $\alpha(\mu; \delta)$ directly but rather consider a related problem associated with another quantity $\tau(\mu)$. The following proposition relates the two quantities.

Proposition 6 *For any $\mu \in \mathcal{D}$ and $\delta \in \mathbb{R}_{>0}$, the stability constant $\alpha(\mu; \delta)$ is bounded from the below by*

$$\alpha(\mu; \delta) \equiv \inf_{v \in \mathcal{V}} \frac{|||v|||_\mu^2}{\|v\|_{\mathcal{W}(\mu;\delta)}^2} \geq \left(1 + \frac{\delta}{\tau_{\mathrm{LB}}(\mu)}\right)^{-1} \equiv \alpha_{\mathrm{LB}}(\mu; \delta),$$

where $\tau_{\mathrm{LB}}(\mu)$ satisfies $\tau_{\mathrm{LB}}(\mu) \leq \tau(\mu) \equiv \inf_{v \in \mathcal{V}} |||v|||_\mu^2 / \|v\|_{\mathcal{V}}^2$.

Proof We note that

$$\frac{1}{\alpha(\mu;\delta)} = \sup_{v\in\mathcal{V}} \frac{\|v\|^2_{\mathcal{W}(\mu;\delta)}}{|||v|||^2_\mu} = \sup_{v\in\mathcal{V}} \frac{|||v|||^2_\mu + \delta\|v\|^2_{\mathcal{V}}}{|||v|||^2_\mu} = 1 + \delta \sup_{v\in\mathcal{V}} \frac{\|v\|^2_{\mathcal{V}}}{|||v|||^2_\mu} = 1 + \frac{\delta}{\tau(\mu)}.$$

Appealing to $\tau_{\mathrm{LB}}(\mu) \le \tau(\mu)$ provides the desired inequality.
We make a few observations. First, if we can provide a lower bound of $\tau(\mu)$, then we can provide a lower bound of $\alpha(\mu;\delta)$. Second, the stability constant is close to unity if we choose $\delta \ll \tau_{\mathrm{LB}}(\mu)$; in particular, the effectivity of $\alpha_{\mathrm{LB}}(\mu;\delta)$ is desensitized from the effectivity of $\tau_{\mathrm{LB}}(\mu)$ as long as $\delta \ll \tau_{\mathrm{LB}}(\mu)$. Third, in the limit of $\delta \to 0$, the stability constant is unity; this is closely related to the complementary variational principle, as discussed in detail in Sect. 4.5. Fourth, the fraction that appears in the definition of $\tau(\mu)$ admits an affine decomposition because $|||v|||^2_\mu \equiv a(v,v;\mu)$ admits an affine decomposition and $\|v\|^2_{\mathcal{V}}$ is parameter independent.

4.4.2 A Residual-Based Lower Bound of the Minimum Eigenvalue

By the Rayleigh quotient, the constant $\tau(\mu)$ is related to the minimum eigenvalue of the following eigenproblem: given $\mu \in \mathcal{D}$, find $(z_i(\mu), \lambda_i(\mu)) \in \mathcal{V} \times \mathbb{R}$ such that

$$a(z_i(\mu), v; \mu) = \lambda_i(\mu)(z_i(\mu), v)_{\mathcal{V}} \quad \forall v \in \mathcal{V} \quad \text{and} \quad \|z_i(\mu)\|_{\mathcal{V}} = 1; \tag{4.9}$$

here the subscript i denotes the index of the eigenpair. We order the eigenpairs in the ascending order of eigenvalues; hence $\tau(\mu) = \min_i \lambda_i(\mu) = \lambda_1(\mu)$.

To compute a lower bound of the minimum eigenvalue, we appeal to Weinstein's method. Towards this end, we introduce the eigenproblem residual associated with any approximate eigenpair $(w,\chi) \in \mathcal{V} \times \mathbb{R}$,

$$r_{\mathrm{eig}}(v; w, \chi; \mu) = a(w, v; \mu) - \chi(w, v)_{\mathcal{V}},$$

and the associated dual norm $\|r_{\mathrm{eig}}(\cdot; w, \chi; \mu)\|_{\mathcal{V}'} \equiv \sup_{v\in\mathcal{V}} \frac{r_{\mathrm{eig}}(v;w,\chi;\mu)}{\|v\|_{\mathcal{V}}}$. The eigenproblem residual is sometimes called the "defect" in the literature. We then introduce the following proposition by Weinstein. (See [3, Chap. 6].)

Proposition 7 *For any $\mu \in \mathcal{D}$ and a pair $(w,\chi) \in \mathcal{V} \times \mathbb{R}$ such that $\|w\|_{\mathcal{V}} = 1$, the distance between χ and the closest eigenvalue is bounded by*

$$\min_i |\lambda_i(\mu) - \chi| \le \|r_{\mathrm{eig}}(\cdot; w, \chi; \mu)\|_{\mathcal{V}'}.$$

Proof See [3, Chap. 6] for a general case or [17] for the specific case.

Corollary 8 *Consider any $\mu \in \mathcal{D}$ and a pair $(w, \chi) \in \mathcal{V} \times \mathbb{R}$ such that $\|w\|_{\mathcal{V}} = 1$. If $|\lambda_1(\mu) - \chi| < |\lambda_2(\mu) - \chi|$, then $\lambda_1(\mu) \geq \chi - \|r_{\mathrm{eig}}(\cdot; w, \chi; \mu)\|_{\mathcal{V}'}$.*

In order to provide a lower bound of the minimum eigenvalue, the corollary requires that the eigenvalue of the approximate eigenpair $(\chi, w) \in \mathcal{V} \times \mathbb{R}$ is closer to $\lambda_1(\mu)$ than to $\lambda_2(\mu)$. *Assuming* this condition is satisfied, we can provide a lower bound of the minimum eigenproblem by bounding the dual norm of the eigenproblem residual, as shown in the following proposition.

Proposition 9 *For any $w \in \mathcal{V}$, $\chi \in \mathbb{R}$, $q \in \mathcal{Q}$, and $\mu \in \mathcal{D}$,*

$$\|r_{\mathrm{eig}}(\cdot; w, \chi; \mu)\|_{\mathcal{V}'} \leq (F_{\mathrm{eig}}(w, \chi, q; \mu))^{1/2} \quad \forall q \in \mathcal{Q},$$

where the bound form is given by

$$F_{\mathrm{eig}}(w, \chi, q; \mu) \equiv \chi^2(\|\chi^{-1}|J(\mu)|Y(\mu)^{-1}EK(\mu)EY(\mu)^{-T}\nabla w - \nabla w - q\|^2_{L^2(\Omega)} + \|w + \nabla \cdot q\|^2_{L^2(\Omega)} + \|w - n \cdot q\|^2_{L^2(\Gamma_N)}). \tag{4.10}$$

Proof The proof is omitted here for brevity. We refer to [17] for a complete proof; unlike the proof of Proposition 5, rigid-body rotation modes do not introduce additional difficulties relative to the scalar case in [17].

We can readily show that for an eigenpair $(z_1(\mu), \lambda_1(\mu)) \in \mathcal{V} \times \mathbb{R}$ of (4.9), $\inf_{q \in \mathcal{Q}} F_{\mathrm{eig}}(z_1(\mu), \lambda_1(\mu), q; \mu) = 0$. Hence, given the exact eigenvalue $\lambda_1(\mu)$, there exists $(w, q) \in \mathcal{V} \times \mathcal{Q}$ such that the lower bound collapses to the exact eigenvalue.

4.4.3 FE Approximation of Bounds of $\tau(\mu)$

Upper Bound An upper bound of $\tau(\mu)$ is readily given by a FE approximation of the eigenproblem (4.9): given $\mu \in \mathcal{D}$, find $(z_1^{\mathcal{N}}(\mu), \lambda_1^{\mathcal{N}}(\mu)) \in \mathcal{V}^{\mathcal{N}} \times \mathbb{R}$ such that

$$a(z_1^{\mathcal{N}}(\mu), v; \mu) = \lambda_1^{\mathcal{N}}(\mu)(z_1^{\mathcal{N}}(\mu), v)_{\mathcal{V}} \quad \forall v \in \mathcal{V} \quad \text{and} \quad \|z_i^{\mathcal{N}}(\mu)\|_{\mathcal{V}} = 1. \tag{4.11}$$

Because $\lambda_1(\mu) \equiv \inf_{v \in \mathcal{V}} |||v|||^2_{\mu}/\|v\|^2_{\mathcal{V}} \leq \inf_{v \in \mathcal{V}^{\mathcal{N}}} |||v|||^2_{\mu}/\|v\|^2_{\mathcal{V}} \equiv \lambda_1^{\mathcal{N}}(\mu)$, we conclude $\tau(\mu) \equiv \lambda_1(\mu) \leq \lambda_1^{\mathcal{N}}(\mu) \equiv \tau^{\mathcal{N}}_{\mathrm{UB}}(\mu)$. We hence set $\tau^{\mathcal{N}}_{\mathrm{UB}}(\mu) \equiv \lambda_1^{\mathcal{N}}(\mu)$.

Lower Bound To compute a lower bound of $\tau(\mu)$ using a FE approximation, we first solve the Galerkin FE problem (4.11) to obtain an approximate eigenpair $(z_1^{\mathcal{N}}(\mu), \lambda_1^{\mathcal{N}}(\mu)) \in \mathcal{V}^{\mathcal{N}} \times \mathbb{R}$. We then solve the minimum bound problem associated with (4.10) for the dual field: given $\mu \in \mathcal{D}$, find $y^{\mathcal{N}}(\mu) \in \mathcal{Q}^{\mathcal{N}}$ such that

$$y^{\mathcal{N}}(\mu) = \arg \inf_{q \in \mathcal{Q}^{\mathcal{N}}} F_{\mathrm{eig}}(z_1^{\mathcal{N}}(\mu), \lambda_1^{\mathcal{N}}(\mu), q; \mu).$$

We then *assume* that $|\lambda_1(\mu) - \lambda_1^{\mathcal{N}}(\mu)| < |\lambda_2(\mu) - \lambda_1^{\mathcal{N}}(\mu)|$ and set

$$\tau_{\mathrm{LB}}^{\mathcal{N}}(\mu) \equiv \lambda_1^{\mathcal{N}}(\mu) - (F_{\mathrm{eig}}(z_1^{\mathcal{N}}(\mu), \lambda_1^{\mathcal{N}}(\mu), y^{\mathcal{N}}(\mu); \mu))^{1/2} \leq \tau(\mu). \tag{4.12}$$

We unfortunately have no means to verify whether the assumption $|\lambda_1(\mu) - \lambda_1^{\mathcal{N}}(\mu)| < |\lambda_2(\mu) - \lambda_1^{\mathcal{N}}(\mu)|$ is satisfied. However, in practice, we have found that smaller eigenvalues of (4.9) are well separated, and the associated eigenfunctions are well approximated even on very coarse meshes. Hence, $\tau_{\mathrm{LB}}^{\mathcal{N}}(\mu)$ defined by (4.12) provides a lower bound of the stability constant $\tau(\mu)$.

4.4.4 Offline-Online Efficient SCM and RB Bounds of $\tau(\mu)$

Lower Bound While the approach described in Sect. 4.4.3 provides a lower bound of the stability constant $\tau(\mu)$ under a plausible assumption, the approach requires FE approximations and is not suited for rapid online evaluation. To overcome the difficultly, we appeal to a version of the successive constraint method (SCM) of Huynh et al. [5] that has been extended to compute a lower bound of the stability constant with respect to an infinite-dimensional function spaces [17]. We refer to [5, 17] for detailed discussion of the algorithm; we here simply present the mechanics for completeness.

For notational simplicity, we first define an operator associated with the bilinear form $a(w, v; \mu)$, $A(\mu) \equiv |J(\mu)|Y(\mu)^{-1}EK(\mu)EY(\mu)^{-T}$. Because $K(\mu)$ and $Y(\mu)^{-1} = I_d \otimes J(\mu)^{-1}$ admit affine decompositions, $A(\mu)$ also admits an affine decomposition, which we denote by $A(\mu) = \sum_{q=1}^{Q_A} \Theta_q^A(\mu) A_q$. The number of terms in the affine expansion Q_A is at most $Q_J Q_{J\mathrm{inv}}^2 Q_K$.

The SCM computes the lower bound as follows. We first introduce a bounding box $B_{Q_A} \equiv \prod_{q=1}^{Q_A} [\hat{\gamma}_q^-, \hat{\gamma}_q^+] \subset \mathbb{R}^{Q_A}$, where $\hat{\gamma}_q^{\pm} \equiv \|\lambda_{\max}(A_q)\|_{L^\infty(\Omega)}$; we can readily evaluate $\|\lambda_{\max}(A_q)\|_{L^\infty(\Omega)}$ since A_q are known. We then define $\mathcal{Y}_{\mathrm{LB},M} \equiv \left\{ y \in B_{Q_A} \mid \sum_{q=1}^{Q_A} \Theta_q^A(\mu') \geq \tau_{\mathrm{LB}}^{\mathcal{N}}(\mu'),\ \forall \mu' \in \Xi_{\mathrm{con}} \right\}$; here $\Xi_{\mathrm{con}} \subset \mathcal{D}$ is a set of judiciously chosen "SCM constraint points" (e.g., by a greedy algorithm) of cardinality M, and $\tau_{\mathrm{LB}}^{\mathcal{N}}(\mu')$, $\mu' \in \Xi_{\mathrm{con}}$, are the FE approximations of lower bound of eigenvalues in (4.12). The SCM lower bound of $\tau(\mu)$ is then given by

$$\tau_{\mathrm{LB},M}(\mu) = \inf_{y \in \mathcal{Y}_{\mathrm{LB},M}} \sum_{q=1}^{Q_A} \Theta_q^A(\mu) y_q. \tag{4.13}$$

We can readily show $\tau_{\mathrm{LB},M}(\mu) \leq \tau(\mu)$; we refer to [5] or [17] for a proof.

The SCM algorithm is online-offline efficient: in the offline stage, we evaluate the constants $\{\gamma_q^{\pm}\}$ by taking the L^∞-norm of A_q and $\{\tau_{\mathrm{LB}}^{\mathcal{N}}(\mu')\}_{\mu' \in \Xi_{\mathrm{con}}}$ by solving $M \equiv |\Xi_{\mathrm{con}}|$ FE problems (4.12); in the online stage, we solve a linear programming problem (4.13) with Q_A variables and M inequality constraints.

Upper Bound While bounding the error in the online stage requires only the lower bound $\tau_{\mathrm{LB},M}(\mu)$, our offline training algorithm also requires a rapidly computable upper bound of $\tau(\mu)$ to select Ξ_{con}. Towards this end, we appeal to a Galerkin RB approximation of $\tau(\mu)$ (c.f. [12]). We introduce a RB space spanned by the eigenfunctions associated with M parameter values: $\mathcal{V}_M^{\mathrm{eig}} = \mathrm{span}\{z_1^{\mathcal{N}}(\mu')\}_{\mu' \in \Xi_{\mathrm{con}}}$. We then solve a RB eigenproblem: given $\mu \in \mathcal{D}$, find $(z_{M,1}(\mu), \lambda_{M,1}(\mu)) \in \mathcal{V}_M^{\mathrm{eig}} \times \mathbb{R}$ such that $\|z_{M,1}(\mu)\|_{\mathcal{V}} = 1$ and

$$a(z_{M,1}(\mu), v; \mu) = \lambda_{M,1}(\mu)(z_{M,1}(\mu), v)_{\mathcal{V}} \quad \forall v \in \mathcal{V}_M^{\mathrm{eig}}. \tag{4.14}$$

Because $\lambda_1(\mu) \equiv \inf_{v \in \mathcal{V}} |||v|||_\mu^2 / \|v\|_{\mathcal{V}}^2 \leq \inf_{v \in \mathcal{V}_M^{\mathrm{eig}}} |||v|||_\mu^2 / \|v\|_{\mathcal{V}}^2 \equiv \lambda_{1,M}(\mu)$, we conclude $\tau(\mu) \equiv \lambda_1(\mu) \leq \lambda_1^{\mathcal{N}}(\mu) \equiv \tau_{\mathrm{UB},M}(\mu)$. We hence set $\tau_{\mathrm{UB},M}(\mu) \equiv \lambda_{M,1}(\mu)$. The RB eigenproblem (4.14) is amenable to offline-online computational decomposition because the form $a(\cdot, \cdot; \mu)$ admits an affine decomposition. In addition, the basis $\mathcal{V}_M^{\mathrm{eig}}$ is generated as a biproduct of computing $\{\tau_{\mathrm{LB}}^{\mathcal{N}}(\mu')\}_{\mu' \in \Xi_{\mathrm{con}}}$ by FE eigenproblem (4.11) in the offline stage.

4.5 Error Bounds

Bounds Having devised offline-online efficient approach for computing an upper bound of the dual norm of the residual and a lower bound of the stability constant, we appeal to Proposition 2 to obtain a computable bound of an energy norm of the error:

$$|||u(\mu) - u_N(\mu)|||_\mu \leq \Delta_N(\mu) \equiv \frac{1}{(\alpha_{\mathrm{LB},M}(\mu;\delta))^{1/2}} (F(u_N(\mu), p_N(\mu); \mu; \delta))^{1/2}.$$

Similarly, we appeal to Proposition 3 to define an approximate compliance output $s_N(\mu) = \ell(u_N(\mu)) + r(u_N(\mu), u_N(\mu); \mu)$ and to provide an error bound

$$|s(\mu) - s_N(\mu)| \leq \Delta_N^s(\mu) \equiv \frac{1}{\alpha_{\mathrm{LB},M}(\mu;\delta)} F(u_N(\mu), p_N(\mu); \mu; \delta).$$

We note that the term $r(u_N(\mu), u_N(\mu); \mu)$ is nonzero because our approximation $u_N(\mu)$ is based on the minimum-bound formulation and not a Galerkin projection.

Complementary Variational Principle There exists a close relationship between our error bound formulation and finite-element error bounds based on the complementary variational principle in, e.g., [6, 9]. If we consider the limit of $\delta \to 0$ for our norm $\|\cdot\|_{\mathcal{W}(\mu;\delta)}$, our bound form (4.7) expressed in the *physical domain* $\tilde{\Omega}(\mu)$

becomes

$$F(w,q;\mu;\delta) = \begin{cases} \|\tilde{C}(\mu)^{1/2}\tilde{q} - \tilde{K}(\mu)^{1/2}\tilde{\nabla}\tilde{w}\|^2_{L^2(\tilde{\Omega})}, & q \in \tilde{\mathcal{Q}}^\star(\mu), \\ \infty, \quad q \notin \mathcal{Q}^\star(\mu), \end{cases}$$

where

$$\tilde{\mathcal{Q}}^\star(\mu) = \{\tilde{q} \in \tilde{\mathcal{Q}}(\mu) \mid -\tilde{\nabla}\cdot\tilde{q} = \tilde{f}(\mu),\ \tilde{n}\cdot\tilde{q} = g(\mu),\ \tilde{q}\text{-tensor is symmetric}\} \tag{4.15}$$

The associated stability constant for $\delta \to 0$ is $\lim_{\delta\to 0}\alpha(\mu;\delta) = 1$.

The conditions that define $\tilde{\mathcal{Q}}(\mu)$ in (4.15) are the dual-feasibility conditions associated with the complementary variational principle. The symmetry of the dual field is a required condition for linear elasticity [9], which is not present for scalar equations. In addition, for $\tilde{q} \in \tilde{\mathcal{Q}}(\mu)$, the complementary variational principle yields $|||\tilde{w}|||^2_\mu \leq \|\tilde{C}(\mu)^{1/2}\tilde{q} - \tilde{K}(\mu)^{1/2}\tilde{\nabla}\tilde{w}\|^2_{L^2(\tilde{\Omega})}$, which implies that the stability constant is unity. Hence, our bound formulation in the limit $\delta \to 0$ is equivalent to the complementary variational principle.

For $\delta > 0$, our approach is a "relaxation" of the complementary variational principle in the sense that it does not require the dual field to lie in the dual-feasible space (4.15). This relaxation facilitates offline-online decomposition, as the construction of the parameter-dependent dual-feasible space $\tilde{\mathcal{Q}}^\star(\mu)$ in an online-efficient manner seems only possible for rather limited cases [15]. However, as a consequence, our stability constant $\alpha(\mu;\delta)$ is not unity, and we require an explicit computation of a lower bound of the stability constant.

4.6 Spatio-Parameter Adaptation

Our spatio-parameter adaptation algorithm for SCM and RB offline training are presented in [17]; we here reproduce the algorithms for completeness.

SCM The SCM training algorithm is shown as Algorithm 1. The algorithm leverages the offline-online efficient upper and lower bounds of τ introduced in Sect. 4.4. In short, the algorithm computes the relative bound gap for each $\mu \in \Xi_{\text{train}}$, identifies μ with the largest bound gap, computes $\tau^{\mathcal{N}}_{\text{UB}}$ and $\tau^{\mathcal{N}}_{\text{UB}}$ to prescribed accuracy $\epsilon_{\text{SCM,FE}}$ using the adaptive FE eigensolver, and updates the SCM constraint set and reduced basis for the eigenproblem. The process is repeated until the bound gap meets ϵ_{SCM} for all $\mu \in \Xi_{\text{train}}$. The two threshold parameters must satisfy $\epsilon_{\text{SCM,FE}} \leq \epsilon_{\text{SCM}} < 1$; in practice we set $\epsilon_{\text{SCM}} \approx 0.8$ and $\epsilon_{\text{SCM,FE}} \leq \epsilon_{\text{SCM,FE}}/2$.

Algorithm 1 Spatio-parameter adaptive SCM training

```
input  : $\Xi_{\rm train} \subset \mathcal{D}$: SCM training set
         $\epsilon_{\rm SCM}$, $\epsilon_{\rm SCM,FE}$: greedy and finite-element bound-gap tolerances
output : $\{\tau^{\mathcal{N}}_{\rm LB}(\mu')\}_{\mu' \in \Xi_{\rm con}}$: SCM constraints
         $\mathcal{V}^{\rm eig}_M = \{z^{\mathcal{N}}_1(\mu')\}_{\mu' \in \Xi_{\rm con}}$: RB eigenproblem space
for $M = 1, 2, \ldots$ do
    Identify the maximum relative $\tau(\mu)$ gap parameter
    $\mu^{(M)} = \arg\sup_{\mu \in \Xi_{\rm train}} (\tau_{{\rm UB},M-1}(\mu) - \tau_{{\rm LB},M-1}(\mu))/\tau_{{\rm UB},M-1}(\mu)$.
    If $\sup_{\mu \in \Xi_{\rm train}} (\tau_{{\rm UB},M}(\mu) - \tau_{{\rm LB},M}(\mu))/\tau_{{\rm UB},M}(\mu) < \epsilon_{\rm SCM}$, terminate.
    Solve (4.11) and (4.12) to obtain eigenpair $(z^{\mathcal{N}}_1(\mu^{(M)}), \lambda^{\mathcal{N}}_1(\mu^{(M)}) \equiv \tau^{\mathcal{N}}_{\rm UB}(\mu^{(M)}))$ and
     a lower bound $\tau^{\mathcal{N}}_{1,{\rm LB}}(\mu)$; invoke mesh adaptivity as necessary such that
     $(\tau^{\mathcal{N}}_{\rm UB}(\mu_M) - \tau^{\mathcal{N}}_{\rm LB}(\mu_M))/\tau^{\mathcal{N}}_{\rm UB}(\mu) < \epsilon_{\rm SCM,FE}$.
    Augment the SCM constraint set, $\Xi_{\rm con} \leftarrow \Xi_{\rm con} \cup \mu^{(M)}$, and update
    $\{\tau^{\mathcal{N}}_{\rm LB}(\mu')\}_{\mu' \in \Xi_{\rm con}}$ and $\mathcal{V}^{\rm eig}_M = \{z^{\mathcal{N}}_1(\mu')\}_{\mu' \in \Xi_{\rm con}}$ accordingly.
end
```

Algorithm 2 Spatio-parameter adaptive RB training

```
input  : $\Xi_{\rm train}$: RB training set
         $\epsilon_{\rm RB}$, $\epsilon_{\rm RB,FE}$: greedy and finite-element error tolerance
output : $\mathcal{V}_N$, $\mathcal{Q}_N$: RB spaces
for $N = 1, 2, \ldots$ do
    Identify the maximum bound parameter $\mu^{(N)} = \arg\sup_{\mu \in \Xi_{\rm train}} \Delta_{N-1}(\mu)$.
    If $\sup_{\mu \in \Xi_{\rm train}} \Delta_{N-1}(\mu) \le \epsilon_{\rm RB}$, terminate.
    Solve (4.8) to obtain FE approximations $u^{\mathcal{N}}(\mu^{(N)})$ and $p^{\mathcal{N}}(\mu^{(N)})$; invoke mesh
     adaptivity as necessary such that $\Delta^{\mathcal{N}}(\mu) \le \epsilon_{\rm RB,FE}$.
    Update RB spaces: $\mathcal{V}_N = {\rm span}\{\mathcal{V}_{N-1}, u^{\mathcal{N}}(\mu^{(N)})\}$ and
    $\mathcal{Q}_N = {\rm span}\{\mathcal{Q}_{N-1}, p^{\mathcal{N}}(\mu^{(N)})\}$.
end
```

RB The RB training algorithm is shown as Algorithm 2. The algorithm leverages the offline-online efficient error bound Δ_N. In short, the algorithm computes the error bound for each $\mu \in \Xi_{\rm train}$, identifies μ with the largest error bound, approximate the solution to prescribed accuracy using the adaptive mixed FE solver, and updates the reduced basis. The process is repeated until the error bound meets $\epsilon_{\rm RB}$ for all $\mu \in \Xi_{\rm train}$. The two threshold parameters must satisfy $\epsilon_{\rm RB,FE} \le \epsilon_{RB}$; in practice we set $\epsilon_{\rm RB,FE} \le \epsilon_{\rm RB}/2$. We set $\delta \equiv \min_{\mu \in \Xi_{\rm train}} \tau_{{\rm LB},M}(\mu)/10$ throughout the training (and in online evaluation); the choice ensures that the stability constant satisfies $10/11 \le \alpha_{{\rm LB},M}(\mu) \le 1$ and in particular is close to unity.

The reduced model constructed by Algorithms 1 and 2 provides an RB approximation $u_N(\mu)$ such that the error $|||u(\mu) - u_N(\mu)|||_\mu$ with respect to the *exact* solution is guaranteed to be less than $\epsilon_{\rm RB}$ for all $\mu \in \Xi_{\rm train}$; for $\mu \notin \Xi_{\rm train}$, the model may yield an approximation with an error greater than $\epsilon_{\rm RB}$, but the approximation is nevertheless equipped with an error bound with respect to the *exact* solution.

4.7 Numerical Results

4.7.1 Problem Description

We consider a linear elasticity problem associated with a cracked square patch of unit-length edges shown in Fig. 4.1. We will refer to the crack embedded in the domain as the "embedded crack" and crack in the center as the "primary crack." Two parameters characterize the embedded crack: the first parameter, $\mu_1 \in [0.25, 0.4]$, controls the vertical location of the crack; the second parameter, $\mu_2 \in [0.3, 0.7]$, controls the length of the crack. The patch is clamped along $\varGamma_D$, is subjected to vertical traction force along $\varGamma_T$, and is traction-free on all other boundaries. The output of interest is compliance.

4.7.2 Uniform Spatio-Parameter Refinement

We first solve the parametrized cracked patch problem using uniform refinement. The spatial meshes are obtained by uniformly refining the initial mesh shown in Fig. 4.2a. The snapshot locations are 2^2, 3^2, 4^2, and 5^2 equispaced points over $\mathcal{D} \equiv [0.25, 0.4] \times [0.3, 0.7]$. All mixed FE discretization is based on $\mathbb{P}^3$ Lagrange and $\mathbb{RT}^2$ Raviart-Thomas elements. For the purpose of assessment, the error bounds are computed on the sampling set $\varXi \subset \mathcal{D}$ consisting of $31 \times 41 = 1271$ equidistributed parameter points.

Figure 4.2b shows the result of the uniform refinement study. On the coarsest mesh with $\mathcal{N} = 1008$ degrees of freedom, the output error bound stagnates for $N \geq 9$ and is of $\mathcal{O}(1)$ independent of the number of snapshots; the error is dominated by the insufficient spatial resolution. Even on the finest mesh with $\mathcal{N} \approx 220{,}000$, the convergence of the error bound is affected by the spatial resolution for $N \geq 16$. This behavior is due to the relatively slow convergence of the FE method in the presence of spatial singularity and a rapid convergence of the RB method for the parametrically smooth problem.

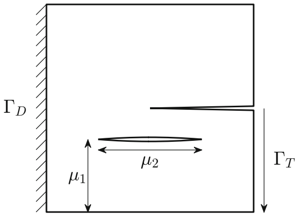


Fig. 4.1 Geometry and parametrization of the cracked patch problem

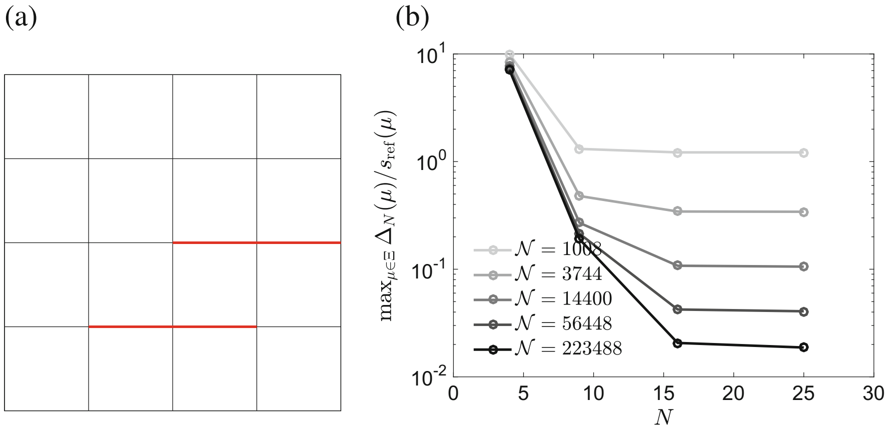


Fig. 4.2 Uniform refinement convergence study: (**a**) initial mesh with the cracks denoted in *red*; (**b**) convergence with N for several FE meshes

4.7.3 Spatio-Parameter Adaptive SCM and RB Refinement

SCM We now apply the spatio-parameter adaptive SCM training, Algorithm 1, using threshold parameters $\epsilon_{\mathrm{SCM}} = 0.8$ and $\epsilon_{\mathrm{SCM,FE}} = 0.2$. Figure 4.3 summarizes the result of the training process. Figure 4.3a shows that the dimension of the adaptive FE space varies from ≈ 3500 to ≈ 7500, depending on the configuration. Figure 4.3b shows that the target maximum relative SCM bound gap of $\epsilon_{\mathrm{SCM}} = 0.8$ is achieved using $M = 40$ constraint points for all $\mu \in \Xi \subset \mathcal{D}$. Figure 4.3c shows that, similar to the original SCM [5], the SCM lower bound of the eigenvalue is rather pessimistic away from the constraint points; as discussed earlier, we accept the pessimistic estimate for the rigor it provides, and in any event the effectivity of the stability constant $\alpha_{\mathrm{LB},M}$ will be desensitized from the pessimistic estimate $\tau_{\mathrm{LB},M}$ thanks to the transformation introduced in Sect. 4.4.1. Figure 4.3d shows that the Galerkin approximation of the upper bound—which in fact approximates very closely the true value of τ—varies smoothly over the parameter domain. The minimum τ_{LB} is bounded from the below by 0.0018; we hence set $\delta = 0.00018$ to ensure that $\alpha_{\mathrm{LB},M}(\mu) > 0.9$.

In order to more closely analyze the adaptive FE approximation of the stability eigenproblem, we show in Fig. 4.4 the adaptation behavior for two configurations associated with the smallest and largest FE spaces. Figure 4.4a–c summarize the behavior for $\mu^{(6)}$, the configuration where the embedded crack is shortest and is far from the primary crack; the final $\mathcal{N} = 3514$ mesh exhibits strong refinement

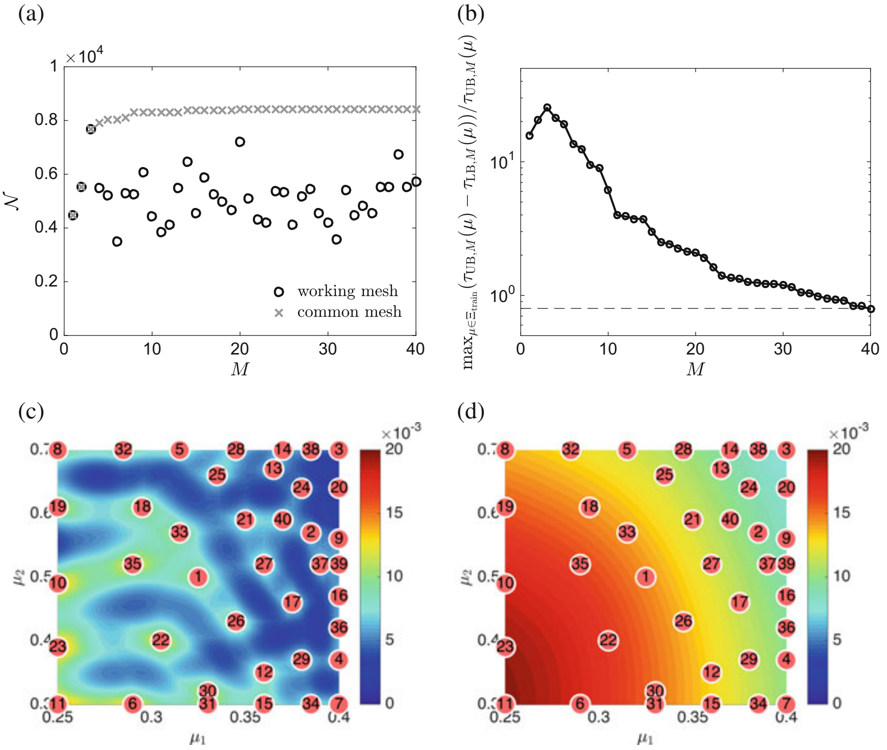


Fig. 4.3 Behavior of the spatio-parameter adaptive greedy method for SCM: (**a**) the dimension of the FE spaces; (**b**) reduction in the bound gap with number of SCM constraints; (**c**) SCM lower bound of τ over $\mathcal{D}$; (**d**) Galerkin reduced-basis upper bound of τ over $\mathcal{D}$

towards the primary crack tip, but relatively weak refinement towards the embedded crack tips. Figure 4.4d–f summarize the behavior for $\mu^{(3)}$, the configuration where the embedded crack is longest and is closest to the primary crack; the final $\mathcal{N} = 7690$ mesh exhibits much stronger refinement towards the embedded crack tips compared to the mesh for $\mu^{(6)}$. As shown in Fig. 4.4c and f, the lower bound is not as effective as the upper bound in general, but we accept the ineffectiveness for the rigor it provides.

RB We now train the RB model using the spatio-parameter adaptive method, Algorithm 2, for threshold parameters $\epsilon_{\rm RB} = 0.01$ and $\epsilon_{\rm RB,FE} = 0.005$. Figure 4.5 summarizes the result of the greedy training. Figure 4.5a shows that the number

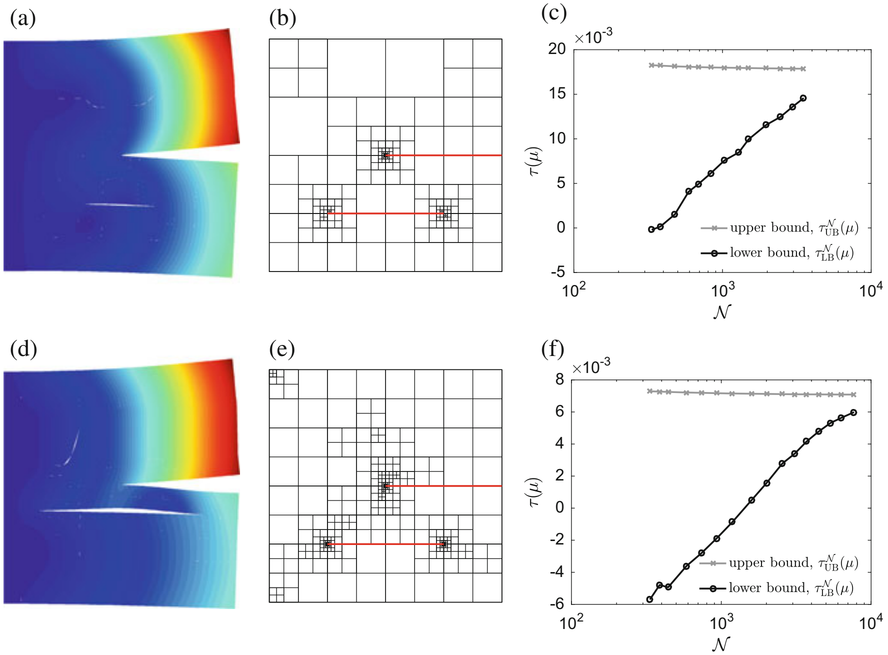


Fig. 4.4 Adaptive FE eigenproblem approximation for (**a**)–(**c**) $\mu^{(6)} = (0.29, 0.3)$ and (**d**)–(**f**) $\mu^{(3)} = (0.4, 0.7)$

of degrees of freedom varies from ≈ 13,000 to ≈ 21,000. Figure 4.5b shows the exponential convergence of the compliance output error with the dimension of the RB space; this is contrary to the behavior for uniform meshes for which the convergence with respect to the parameter dimension is limited by the insufficient spatial resolution. Figure 4.5c shows that reduced model produces an error less than $\epsilon_{\mathrm{RB}} = 10^{-2}$ for any parameter value in $\mathcal{D}$ (or more precisely at least Ξ). Figure 4.5d shows that the final common mesh which reflects refinement required for all configurations over $\mathcal{D}$ exhibits strong refinement towards the crack tips and some corners.

As we have done for the eigenproblem, we show in Fig. 4.6 the adaptive FE solution for two configurations associated with the smallest and largest FE spaces. Figure 4.6a–c summarize the behavior for $\mu^{(17)}$, the configuration where the embedded crack is shortest and far from the primary crack; the final $\mathcal{N} = 13{,}270$ mesh shows relatively weak refinement towards the embedded crack tips. Figure 4.6d–f summarize the behavior for $\mu^{(2)}$, the configuration where the embedded crack

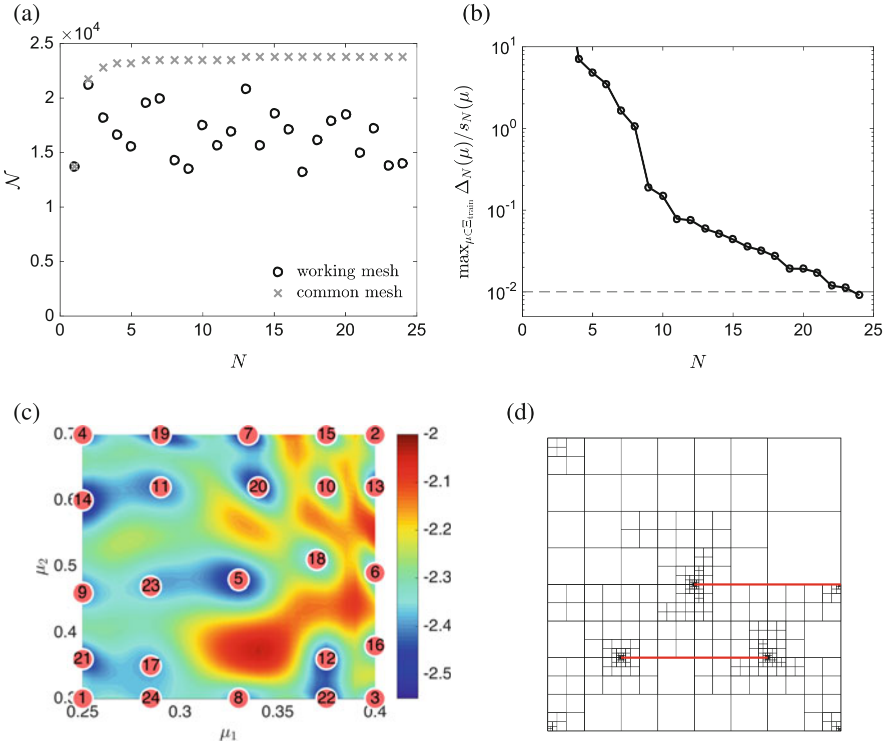


Fig. 4.5 Behavior of the spatio-parameter adaptive RB generation: (**a**) the dimension of the FE spaces; (**b**) reduction in the error bound with the dimension of RB space; (**c**) output error bound over $\mathcal{D}$; (**d**) final common mesh

is longest and closest to the primary crack; we observe much stronger refinement towards all crack tips. For both cases, the effectivity of the compliance output error bound is less than 10, which is acceptable given that this is (rigorous) bounds of the error in the outputs. For assessment purpose, the reference output is computed using an adaptive FE method with an error tolerance that is ten times tighter than the target tolerance.

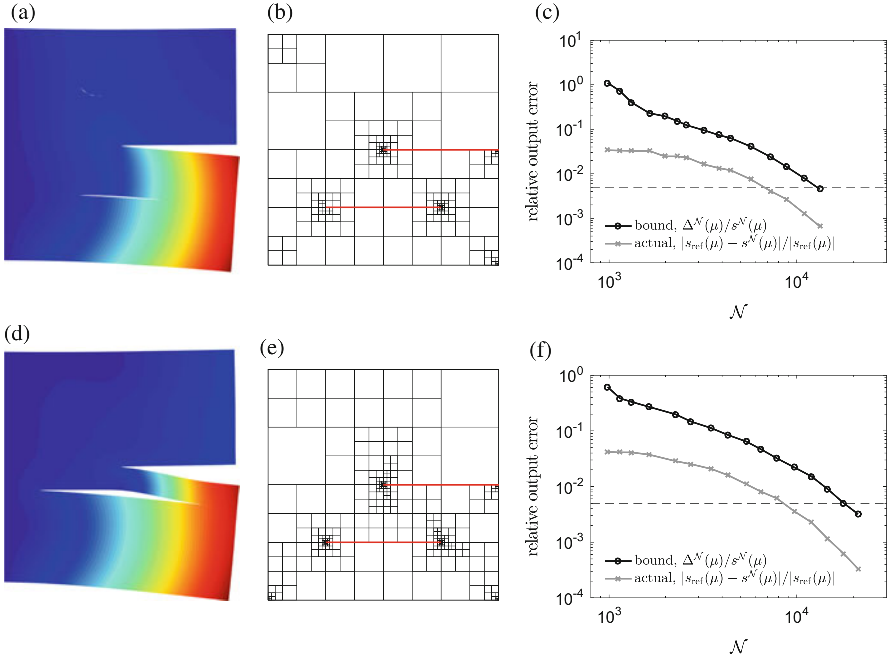


Fig. 4.6 Adaptive FE approximation for (**a**)–(**c**) $\mu^{(17)} = (0.285, 0.35)$ and for (**d**)–(**f**) $\mu^{(2)} = (0.4, 0.7)$

Acknowledgements I would like to thank Prof. Anthony Patera of MIT for the many fruitful discussions. This work was supported by OSD/AFOSR/MURI Grant FA9550-09-1-0613, ONR Grant N00014-11-1-0713, and the University of Toronto Institute for Aerospace Studies (UTIAS).

References

1. Ali, M., Steih, K., Urban, K.: Reduced basis methods based upon adaptive snapshot computations. Adv. Comput. Math. **43**, 257–294 (2017)
2. Brenner, S.C., Scott, L.R.: The Mathematical Theory of Finite Element Methods, 3rd edn. Springer, New York (2008)
3. Chatelin, F.: Spectral Approximations of Linear Operators. Academic, New York (1983)
4. Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for stress intensity factors. Int. J. Numer. Methods Eng. **72**(10), 1219–1259 (2007)
5. Huynh, D.B.P., Rozza, G., Sen, S., Patera, A.T.: A successive constraint linear optimization method for lower bounds of parametric coercivity and inf-sup stability constants. C.R. Acad. Sci. Paris, Ser. I **345**, 473–478 (2007)
6. Ladevèze, P., Leguillion, D.: Error estimate procedure in the finite element method and applications. SIAM J. Numer. Anal. **20**, 485–509 (1983)
7. Milani, R., Quarteroni, A., Rozza, G.: Reduced basis method for linear elasticity problems with many parameters. Comput. Methods Appl. Mech. Eng. **197**, 4812–4829 (2008)

8. Ohlberger, M., Schindler, F.: Error control for the localized reduced basis multi-scale method with adaptive on-line enrichment. SIAM J. Sci. Comput. **37**, A2865–A2895 (2015)
9. Parés, N., Bonet, J., Huerta, A., Peraire, J.: The computation of bounds for linear-functional outputs of weak solutions to the two-dimensional elasticity equations. Comput. Methods Appl. Mech. Eng. **195**, 430–443 (2006)
10. Phuong, H.D.B.: Reduced basis approximation and application to fracture problems. Ph.D. thesis, Singapore-MIT Alliance, National University of Singapore (2007)
11. Raviart, P.A., Thomas, J.M.: A mixed finite element method for 2nd order elliptic problems. In: Lecture Notes in Mathematics, vol. 606, pp. 292–315. Springer, Berlin (1977)
12. Rovas, D.V.: Reduced-basis output bound methods for parametrized partial differential equations. Ph.D. thesis, Massachusetts Institute of Technology (2003)
13. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and *a posteriori* error estimation for affinely parametrized elliptic coercive partial differential equations — application to transport and continuum mechanics. Arch. Comput. Methods Eng. **15**(3), 229–275 (2008)
14. Veroy, K.: Reduced-basis methods applied to problems in elasticity: analysis and applications. Ph.D. thesis, Massachusetts Institute of Technology (2003)
15. Yano, M.: A reduced basis method with exact-solution certificates for steady symmetric coercive equations. Comput. Methods Appl. Mech. Eng. **287**, 290–309 (2015)
16. Yano, M.: A minimum-residual mixed reduced basis method: exact residual certification and simultaneous finite-element and reduced-basis refinement. Math. Model. Numer. Anal. **50**, 163–185 (2016)
17. Yano, M.: A reduced basis method for coercive equations with an exact solution certificate and spatio-parameter adaptivity: energy-norm and output error bounds (2016, submitted)

Chapter 5
A Reduced Basis Method for Parameter Functions Using Wavelet Approximations

Antonia Mayerhofer and Karsten Urban

Abstract We consider parameterized parabolic partial differential equations (PDEs) with variable initial conditions, which are interpreted as a parameter function within the Reduced Basis Method (RBM). This means that we are facing an infinite-dimensional parameter space. We propose to use the space-time variational formulation of the parabolic PDE and show that this allows us to derive a two-step greedy method to determine offline separately the reduced basis for the initial value and the evolution. For the approximation of the initial value, we suggest to use an adaptive wavelet approximation. Online, for a given new parameter function, the reduced basis approximation depends on its (quasi-)best N-term approximation in terms of the wavelet basis. A corresponding offline-online decomposable error estimator is provided. Numerical experiments show the flexibility and the efficiency of the method.

5.1 Introduction

The reduced basis method (RBM) is a well-known model reduction method for parameterized partial differential equations (PDE) within multi-query and/or realtime context situations. Of course, the structure and the dimension of the parameter set $\mathscr{D}$ has significant influence on the efficiency of any RBM. Typically, one has $\mathscr{D} \subset \mathbb{R}^P$ with P being "reasonably" small. In some applications, however, P may be large, even infinite. Such a case occurs if one faces a parameter *function* so that $\mathscr{D} \subset H$ with H being a function (Hilbert) space of infinite dimension.

We consider a particularly relevant class of problems involving parameter functions, namely parabolic problems with variable initial conditions, i.e., we consider the initial condition as a parameter. One possible application are pricing

A. Mayerhofer
Ulm University, Institute for Mathematical Finance, Helmholzstr. 18, 89081 Ulm, Germany
e-mail: antonia.mayerhofer@uni-ulm.de

K. Urban (✉)
Ulm University, Institute for Numerical Mathematics, Helmholtzstr. 20, 89081 Ulm, Germany
e-mail: karsten.urban@uni-ulm.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_5

and hedging problems in finance, where the payoff is used as a parameter, see [4, 5]. Using a parameterized initial value in an evolution equation is a challenge by itself for standard RBM-techniques such as the POD-Greedy method [3], since usually the error is propagating over time.

This is the reason why we rely on the space-time variational formulation for parabolic problems introduced in [6] and being used within the RB-framework e.g. in [9, 10]. Within this framework, we show that we can separate the treatment of the initial value parameter function from additional parameters the PDE may have. We introduce a two-stage Greedy method for computing a corresponding reduced basis in the offline phase.

The issue remains how to approximate a parameter *function*, an ∞-dimensional object (expansion in a separable Hilbert space). We propose an adaptive wavelet approximation for the parameter function online. We show that several ingredients for the RB approximation can be precomputed in the offline phase and how to realize an online-efficient approximation for a new parameter function. Numerical results show the flexibility, efficiency and the approximation quality of the proposed method.

The remainder of this chapter is organized as follows. In Sect. 5.2, we recall those main facts of the RBM that we need here. Section 5.3 is devoted to a brief survey of the space-time variational formulation for parabolic problems as well as its parametric variant. Our suggested RBM for problems with parameter functions is detailed in Sect. 5.4 also including the description of the use of an adaptive wavelet approximation. We report our numerical results in Sect. 5.5 and finish with some conclusions in Sect. 5.6. We refer to [4] to more details on the presented material.

5.2 Reduced Basis Method

Let $\mathscr{D}$ be some parameter space. Consider the parametrized PDE

$$\text{find } u \equiv u(\mu) \in X: \quad b(u, v; \mu) = f(v; \mu) \quad \forall\, v \in Y; \quad \mu \in \mathscr{D}, \tag{5.1}$$

where $b : X \times Y \times \mathscr{D} \to \mathbb{R}$ is a parameter-dependent bilinear form and $f : Y \times \mathscr{D} \to \mathbb{R}$ a parameter-dependent linear form. We assume well-posedness of (5.1). For discrete (but high-dimensional) trial $X^{\mathscr{N}} \subset X$ and test $Y^{\mathscr{N}} \subset Y$ spaces with $\dim(X^{\mathscr{N}}) = \dim(Y^{\mathscr{N}}) = \mathscr{N} \gg 0$ and every $\mu \in \mathscr{D}$, an associated $\mathscr{N}$-dimensional (detailed) linear system has to be solved that is given by

$$\text{find } u^{\mathscr{N}} \equiv u^{\mathscr{N}}(\mu) \in X^{\mathscr{N}}: \quad b(u^{\mathscr{N}}, v; \mu) = f(v; \mu) \quad \forall v \in Y^{\mathscr{N}}. \tag{5.2}$$

We assume well-posedness and uniform stability w.r.t. $\mathscr{N}$ for this detailed system.

In a multi-query or real-time context, the solution of this detailed system (sometimes called "truth") is often too costly. In order to reduce the system, we consider the solution subset ("manifold") $M(\mathscr{D}) = \{u(\mu^{\mathscr{N}}) \in X^{\mathscr{N}} : \mu \in \mathscr{D}\}$. The

RBM aims to approximate $M(\mathscr{D})$ by a lower dimensional space $X_N \subset X^{\mathscr{N}}$ where, $\dim(X_N) = N \ll \mathscr{N}$. The reduced problem then reads

$$\text{find } u_N \equiv u_N(\mu) \in X_N : \quad b(u_N, v; \mu) = f(v; \mu) \quad \forall\, v \in Y_N \tag{5.3}$$

with an appropriate test space Y_N (which may also be parameter-dependent, i.e., $Y_N(\mu)$). Hence, a linear system of dimension N needs to be solved.

The RBM is divided into an offline and an online phase. In the offline phase, bases spanning the reduced system X_N, Y_N are generated by computing detailed solutions $u(\mu^i) \in X^{\mathscr{N}}$ for a well-chosen sample set of parameters $\mu^1, \dots, \mu^N \subset \mathscr{D}$ along with a reduced stable test space Y_N. The reduced system is then solved online for new values of the parameters $\mu \in \mathscr{D}$. The goal is that the reduced system is *online-efficient*, which means it can be solved with an amount of work independent of the detailed dimension $\mathscr{N}$.

In order to reach the latter goal, a standard assumption is to require that the forms b and f are decomposable w.r.t. the parameter (sometimes called "affine decomposition"), i.e., there exist $Q_b, Q_f \in \mathbb{N}$ and functions $\theta_q^b, \theta_q^f : \mathscr{D} \to \mathbb{R}$ such that

$$b(u, v; \mu) = \sum_{q=1}^{Q_b} \theta_q^b(\mu) b_q(u, v), \quad f(v; \mu) = \sum_{q=1}^{Q_f} \theta_q{}^f(\mu) f_q(v) \tag{5.4}$$

with (bi-)linear forms b_q and f_q independent of μ. First, the reduced trial functions $u^i := u^{\mathscr{N}}(\mu^i) \in X^{\mathscr{N}}$ and corresponding inf-sup-stable test functions $v^i \in Y^{\mathscr{N}}$, $i = 1, \dots, N$, are computed. Then, the parameter-independent components are computed and stored, e.g.

$$b_q(u^i, v^j) = \sum_{k,l=1}^{\mathscr{N}} \alpha_{i;k}\, \tilde{\alpha}_{j;l}\, b_q(\varphi_k^{\mathscr{N}}, \tilde{\varphi}_l^{\mathscr{N}}),$$

where $\varphi_k^{\mathscr{N}}$, $\tilde{\varphi}_l^{\mathscr{N}}$ are the basis functions of the detailed spaces $X^{\mathscr{N}}$, $Y^{\mathscr{N}}$ and $\alpha_{i;k}$, $\tilde{\alpha}_{j;l}$ are the expansion coefficients of the reduced basis functions u^i, v^j in terms of the detailed basis functions. The linear forms $f_q(v^j)$ are precomputed in a similar fashion. In the online phase, only $\theta_q^b(\mu)$ and $\theta_q^f(\mu)$ need to be evaluated for a new parameter μ and the sums in (5.4) can be computed with complexity independent of $\mathscr{N}$, i.e., online-efficient.

5.3 Space-Time Variational Formulation for Parabolic Problems

We follow e.g. [6, 9, 10] for the introduction of space-time variational formulations for parabolic PDEs in terms of Bochner(-Lebesgue) spaces. Let $H \hookrightarrow V$ be densely embedded Hilbert spaces. By identifying H with its dual H', we obtain the Gelfand triple $V \hookrightarrow H \hookrightarrow V'$, i.e., the scalar product $(\cdot,\cdot)_H$ on H generates the duality pairing $\langle\cdot,\cdot\rangle_{V'\times V}$. We denote the induced norms on V and H by $|\cdot|_V$ and $|\cdot|_H$, respectively and seek the solution in the space

$$\mathbb{X} := \{u \in L_2(I;V) : \dot{u} \in L_2(I;V')\}, \quad I := (0,T) \subset \mathbb{R},$$

equipped with the graph norm $\|u\|_{\mathbb{X}}^2 := \|u\|_{L_2(I;V)}^2 + \|\dot{u}\|_{L_2(I;V')}^2$.

The parameter spaces are assumed to be of the form $\mathscr{D} = \mathscr{D}_0 \times \mathscr{D}_1 \subset H \times \mathbb{R}^P$, where $\mu_0 \in \mathscr{D}_0$ accounts for the initial value (i.e., a function) and $\mu_1 \in \mathscr{D}_1$ for parameters in the PDE. To be more specific, consider a parameter-dependent bilinear form $a : V \times V \times \mathscr{D}_1 \to \mathbb{R}$ with induced linear operator $\mathscr{A}(\mu_1) \in \mathscr{L}(V,V')$ as $\langle \mathscr{A}(\mu_1)\phi, \psi\rangle_{V'\times V} = a(\phi,\psi;\mu_1)$ for $\phi,\psi \in V$.[1] Then, given a non-parametric right-hand side $g \in L_2(I;V')$, we seek $u(t) \in V$, $t \in I$ such that for $\mu = (\mu_0,\mu_1) \in \mathscr{D}$

$$\dot{u}(t) + \mathscr{A}(\mu_1)u(t) = g(t) \quad \text{in } V', t \in I \text{ a.e.}, \qquad u(0) = \mu_0 \quad \text{in } H. \tag{5.5}$$

We assume that there exist constants $C_a(\mu_1) > 0$, $\alpha_a(\mu_1) > 0$ and $\lambda_a(\mu_1) \in \mathbb{R}$ such that for all $\phi,\psi \in V$

$$|a(\phi,\psi;\mu_1)| \le C_a(\mu_1)|\phi|_V|\psi|_V \qquad \text{(continuity)}, \tag{5.6}$$

$$a(\phi,\phi;\mu_1) + \lambda_a(\mu_1)|\phi|_H^2 \ge \alpha_a(\mu_1)|\phi|_V^2 \qquad \text{(Gårding inequality)}. \tag{5.7}$$

Remark 1

(i) For $u \in \mathbb{X}$ the initial condition in (5.5) is meaningful since $\mathbb{X} \hookrightarrow \mathscr{C}(\bar{I};H)$ [7, III. Proposition 1.2].

(ii) We assume that g is parameter-independent just for ease of presentation. All what is said here extends to parameter-dependent right-hand sides as well.

The test space is chosen as $\mathbb{Y} := L_2(I;V) \times H$ equipped with the graph norm $\|v\|_{\mathbb{Y}}^2 = \|v_1\|_{L_2(I,V)}^2 + |v_2|_H^2$, $v = (v_1,v_2) \in \mathbb{Y}$. For $w \in \mathbb{X}$, $v = (z,h) \in \mathbb{Y}$ and $\mu \in \mathscr{D}$, we define

$$b(w,v;\mu) := \int_I \langle \dot{w}(t), z(t)\rangle_{V'\times V}\, dt + \int_I a(w(t),z(t);\mu_1)dt + (w(0),h)_H, \tag{5.8}$$

$$f(v;\mu) := \int_I \langle g(t), z(t)\rangle_{V'\times V}\, dt + (\mu_0,h)_H. \tag{5.9}$$

The variational formulation of the parameterized parabolic PDE is then given by

$$\text{find } u \in \mathbb{X} : \quad b(u,v;\mu) = f(v;\mu) \quad \forall\, v \in \mathbb{Y}. \tag{5.10}$$

[1] Note, that we also allow for time-dependent bilinear forms, i.e., non-LTI problems as e.g. in [6].

Note, that (5.10) and (5.5) are in fact equivalent. Well-posedness was shown e.g. in [6, Theorem 5.1]. The linear operator $\mathscr{B}(\mu) : \mathbb{X} \to \mathbb{Y}'$ induced by $\langle \mathscr{B}(\mu)u, v\rangle := b(u, v; \mu)$, $v \in \mathbb{Y}$, is thus injective, which implies the inf-sup condition with a lower inf-sup-bound β_{LB}, i.e.,

$$\inf_{u\in\mathbb{X}} \sup_{v\in\mathbb{Y}} \frac{b(u, v; \mu)}{\|u\|_{\mathbb{X}}\|v\|_{\mathbb{Y}}} \geq \beta(\mu) \geq \beta_{\mathrm{LB}} > 0. \tag{5.11}$$

The lower inf-sup-bound plays a major role in the a-posteriori error estimation of the space-time RBM.

5.4 Reduced Basis Method for Parameter Functions

Recall, that the parameter $\mu_0 \in \mathscr{D}_0$ is the initial value, i.e., a parameter function in an infinite-dimensional parameter set (function space). We are now going to introduce an approach to deal with this challenge.

5.4.1 Using the Initial Value as Parameter in a Space-Time Setting

We start by separating the (bi-)linear forms in (5.8), (5.9):

$$\begin{aligned} b(w, v; \mu) &= \int_I \langle \dot{w}(t), z(t)\rangle_{V'\times V}\, dt + \int_I a(w(t), z(t); \mu_1) dt + (w(0), h)_H \qquad (5.12)\\ &=: b_1(w, z; \mu_1) + (w(0), h)_H \end{aligned}$$

$$f(v; \mu) = \int_I \langle g(t), z(t)\rangle_{V'\times V}\, dt + (\mu_0, h)_H =: g_1(z) + (\mu_0, h)_H \tag{5.13}$$

for $\mu = (\mu_0, \mu_1) \in \mathscr{D}$, $w \in \mathbb{X}$ and $(z, h) \in \mathbb{Y}$. Note, that $b(\cdot, \cdot; \mu)$ only depends on μ_1, whereas $f(\cdot; \mu)$ only depends on μ_0, the latter one just for convenience.

The detailed (truth) discretization is induced by $\mathbb{X}^{\mathscr{N}} \subset \mathbb{X}$ and $\mathbb{Y}^{\mathscr{N}} \subset \mathbb{Y}$ with $\dim(\mathbb{X}^{\mathscr{N}}) = \dim(\mathbb{Y}^{\mathscr{N}}) = \mathscr{N}$. We assume well-posedness and uniform stability w.r.t. $\mathscr{N}$ of the truth problem [4]. Note, that $\mathbb{X} = H^1(I) \otimes V$ and $\mathbb{Y} = L_2(I) \otimes V \times H$ are tensor products, so that it is convenient to construct the detailed spaces accordingly,

$$\begin{aligned} \mathbb{X}^{\mathscr{N}} &= (E_0^1 \oplus E_1^{\mathscr{K}}) \otimes V^{\mathscr{J}} = (E_0^1 \otimes V^{\mathscr{J}}) \oplus (E_1^{\mathscr{K}} \otimes V^{\mathscr{J}}) =: Q^{\mathscr{J}} \oplus W^{\mathscr{L}},\\ \mathbb{Y}^{\mathscr{N}} &= (F^{\mathscr{K}} \otimes V^{\mathscr{J}}) \times V^{\mathscr{J}} =: Z^{\mathscr{L}} \times V^{\mathscr{J}}, \quad \dim(W^{\mathscr{L}}) = \mathscr{L} := \mathscr{J}\mathscr{K}. \end{aligned}$$

Here, E_0^1 contains the temporal basis function τ^0 (say) at $t = 0$ ($\dim(E_0^1) = 1$), $E_1^{\mathscr{K}}$ collects the remaining basis functions of the ansatz space in time[2] and $F^{\mathscr{K}}$ consists of the $\mathscr{K}$ temporal basis functions of the test space, see e.g. [10]. All superscripts indicate the dimension of the spaces, so that $\mathscr{N} := \dim(\mathbb{X}^{\mathscr{N}}) = \mathscr{J} + \mathscr{L} = \dim(\mathbb{Y}^{\mathscr{N}})$. This discretization allows a two-step computation for $\mu = (\mu_0, \mu_1) \in \mathscr{D}$ as follows:

$$\text{(a) Find } q(\mu_0) \in V^{\mathscr{J}}: \quad (q(\mu_0), h)_H = (\mu_0, h)_H \qquad \forall h \in V^{\mathscr{J}}, \tag{5.14}$$

$$\text{(b) Find } w(\mu) \in W^{\mathscr{L}}: \quad b_1(w, z; \mu_1) = f_1(z; \mu_1, \tau^0 \otimes q(\mu_0)) \quad \forall z \in Z^{\mathscr{L}}, \tag{5.15}$$

with b_1 as in (5.12) and $f_1(z; \mu_1, w) := g_1(z) - b_1(w, z; \mu_1)$ with g_1 as in (5.13).

The space-time variational approach allows us to use the standard RB-setting in Sect. 5.2 for the a posteriori error estimate in terms of the residual. It turns out that the separation in (5.14) is also crucial here. Let $u_N(\mu) \in \mathbb{X}_N \subset \mathbb{X}^{\mathscr{N}}$ (the RB-approximation to be detailed below), then we get for any $v = (z, h) \in \mathbb{Y}^{\mathscr{N}}$

$$\begin{aligned} r_N(v; \mu) &= f(v; \mu) - b(u_N(\mu), v; \mu) \\ &= g_1(z; \mu_1) - b_1(u_N(\mu), z; \mu_1) + (\mu_0 - u_N(0; \mu), h)_H \\ &=: r_{N,1}(z; \mu) + r_{N,0}(h; \mu). \end{aligned} \tag{5.16}$$

This separation of the residual allows us to control the error for the initial value ($t = 0$) and the evolution separately.

5.4.2 Wavelet Approximation for the Parameter Function

The initial value is a function in $\mathscr{D}_0 \subseteq H$. In principle, we could use any stable basis in H to represent the initial value. However, as indicated in Sect. 5.1, we do not want to fix any possible representation of μ_0 a priori as in [5], but adapt it during the online phase. Hence, we need a basis for H that allows for a rapid and local update of a given new μ_0. We have chosen wavelets. A detailed introduction to wavelets goes far beyond the scope of the present paper, we thus refer e.g. to [8] for details and sketch here just those ingredients that are particularly relevant in the RB-context.

[2] If we use a function in space, say $q \in V^{\mathscr{J}}$, as initial value, we "embed" it into $Q^{\mathscr{J}}$, i.e., we set $\tau^0 \otimes q \in Q^{\mathscr{J}}$ with the temporal basis function τ^0 at $t = 0$.

Wavelets on the real line are usually formed via translation and dilation of a single (compactly supported) function $\psi : \mathbb{R} \to \mathbb{R}$, often called *mother wavelet*, i.e.,

$$\psi_\lambda(x) := 2^{j/2}\psi(2^j x - k), \quad x \in \mathbb{R}, \qquad j \in \mathbb{N}_0 \,(\text{the } \mathit{level}),\ k \in \mathbb{Z},\ \lambda = (j,k).$$

The simplest example is the *Haar wavelet*, where $\psi(x)^{\text{Haar}} := \begin{cases} 1, & 0 \le x < 0.5, \\ -1, & 0.5 \le x < 1. \end{cases}$ Then $\boldsymbol{\Psi}^{\text{Haar}} := \{\psi_\lambda^{\text{Haar}} : \lambda \in \mathbb{Z} \times \mathbb{Z}\}$ is an orthonormal basis (ONB) for $L_2(\mathbb{R})$. If one only uses $\{(j,k) : j \in \mathbb{N}_0,\ 0 \le k < 2^j\}$ instead of $\mathbb{Z} \times \mathbb{Z}$, an ONB of $L_2(0,1)$ results. We denote such general index sets by Λ.

Definition 1 A countable set $\Psi := \{\psi_\lambda : \lambda \in \Lambda\} \subset H$ is called *wavelet basis*, if

(i) Ψ is a *Riesz basis* for H, i.e., there exist constants $0 < c_\Psi \le C_\Psi < \infty$ such that

$$c_\Psi \sum_{\lambda \in \Lambda} |d_\lambda|^2 \le \left| \sum_{\lambda \in \Lambda} d_\lambda \psi_\lambda \right|_H^2 \le C_\Psi \sum_{\lambda \in \Lambda} |d_\lambda|^2, \text{ (in short: } \|\boldsymbol{d}\|_{\ell_2} \sim \|\boldsymbol{d}^T \Psi\|_H\text{);} \tag{5.17}$$

(ii) Ψ has *local support*, i.e., $|\text{supp}(\psi_\lambda)| \sim 2^{-|\lambda|}$, $\lambda = (j,k)$, $|\lambda| := j$;

(iii) Ψ has $\tilde{d}$ *vanishing moments*, i.e., $((\cdot)^p, \psi_\lambda)_H = 0$ for all $0 \le p < \tilde{d}$ and $|\lambda| > 0$.

Remark 2 The Riesz representation theorem yields the existence of a *dual* wavelet basis $\tilde{\Psi}$ with the same properties but a possibly different d instead of $\tilde{d}$ in (iii). Typically, ψ_λ is a piecewise polynomial of degree d.

Nowadays, there is a whole variety of wavelet systems available, on general domains $\Omega \subset \mathbb{R}^n$, with arbitrary smoothness and additional properties. Definition 1 has several consequences. Just to mention a few (and without going into detail due to page restrictions):

1. The wavelet coefficients d_λ decay fast with increasing level $|\lambda|$.
2. The wavelet coefficients are small in regions where the function is smooth—they indicate regions of local non-smoothness (like isolated singularities).
3. The *norm equivalence* (5.17) can be extended to scales of Sobolev spaces H^s around $s = 0$ and allow for an online-efficient computation of the residual [1, 2].
4. There are fast adaptive methods to approximate a given function in H (to be applied to approximate μ_0 online). Roughly speaking, one only needs higher level wavelets in regions where the function is locally non-smooth. The norm equivalence (5.17) ensures that this procedure converges very fast (see below).

Remark 3 It might be a first naive idea to choose a finite subset of wavelet-indices, say $\Lambda_M \subset \Lambda$, and use the corresponding expansion coefficients as a parameter set of dimension $|\Lambda_M|$. Of course, this severely limits the choice of initial conditions. If one is interested in a whole variety of such functions including localized effects, a sufficient approximation causes $|\Lambda_M|$ to be huge—thus making this idea computationally infeasible. Only if one has some knowledge on the shape of the initial value, one may a-priori fix the approximation space, see e.g. [4, 5].

5.4.3 Online-Offline Reduced Basis Method

First Step Greedy: Adaptive Approximation of the Initial Value We start by collecting offline-information for the online initial value approximation given a new parameter function μ_0. We collect those information in a library $\mathbb{L}_{\text{init}}$ that is computed offline. Of course, there are several possibilities how to construct such a library, i.e., which kind of a priori knowledge is used. The less information is used and/or the more flexibility to approximate μ_0 is required, the larger $\mathbb{L}_{\text{init}}$ will be—at the expense of offline cost. On the other hand, additional information may significantly reduce such cost. In this paper, we introduce a rather flexible (but offline costly) method which is solely based upon the goal of achieving a sufficiently good approximation. For alternatives, we refer to [4].

To this end, the standard Jackson-estimate allows to truncate a wavelet expansion at a certain maximum level and to control the approximation error in terms of the Sobolev regularity. In fact, if we denote by S_j the subspace of H generated by all wavelets with level $|\lambda| \leq j$, then

$$\inf_{v_j \in S_j} |v - v_j|_{L_2(\Omega)} \leq C\, 2^{-js} |v|_{H^s(\Omega)}, \quad v \in H^s(\Omega),\ s \leq d,$$

with d as in Remark 2 [8, (5.30)]. Hence, if we know (or fix) the regularity of all candidates for the initial values as well as an estimate for C, we may fix a maximal level, say J. For all wavelets in the corresponding space S_J, we precompute the associated "snapshot" $q^\lambda \in Q^{\mathscr{I}}$ for the initial value by solving

$$(q^\lambda, \phi)_H = (\psi_\lambda, \phi)_H \quad \forall \phi \in S_J,\ |\lambda| \leq J. \tag{5.18}$$

We thus obtain in the order of 2^J such initial value snapshots and store them in an *Initial Value Library* $\mathbb{L}_{\text{init}} = \{q^\lambda\ : \lambda \in \Lambda, |\lambda| \leq J\}$.

Second Step Greedy: Evolution Snapshots Based upon some library $\mathbb{L}_{\text{init}}$, we precompute snapshots for the corresponding evolutions. This is done by an adapted standard Greedy scheme w.r.t. some training set $\mathscr{D}_1^{\text{train}} \subset \mathscr{D}_1$. The arising *Evolution Greedy* is detailed in Algorithm 1. Note, that the training phase is performed for each wavelet index λ with the respective training set $\{\psi_\lambda\} \times \mathscr{D}_1^{\text{train}}$. For each wavelet index λ, Algorithm 1 produces an RB space of the form $W^\lambda_{N(\lambda)} = \text{span}\{w^1_\lambda, \dots, w^{N(\lambda)}_\lambda\}$ of dimension $N(\lambda)$, where w^i_λ solves the space-time parabolic problem

$$b_1(w^i_\lambda, z; \mu^i_1) = g_1(z) - b_1(q^\lambda, z; \mu^i_1) \quad \forall\, z \in \mathbb{Y}^{\mathscr{N}} \tag{5.19}$$

Algorithm 1 Evolution greedy

```
Require: training set $\mathscr{D}^{\text{train}} = \{\psi_\lambda\} \times \mathscr{D}_1^{\text{train}} \subset \mathscr{D}$, tolerance $\text{tol}_1 > 0$
Choose $\mu^1 \in \mathscr{D}^{\text{train}}$, $\mu^1 := (\psi_\lambda, \mu_1^1)$. Get precomputed $q^\lambda \in \mathbb{L}_{\text{init}}$
Compute $w(\mu^1) \in W^{\mathscr{L}}$ as in (5.19), $\Xi_1^\lambda = \{w(\mu^1)\}$
for $\ell = 1, \dots, N^{\max}$ do
    $\mu^{\ell+1} = \arg \max_{\mu \in \mathscr{D}^{\text{train}}} \Delta_\ell^1(\mu)$
    if $\Delta_\ell^1(\mu^{\ell+1}) < \text{tol}_1$ then $N(\lambda) := \ell$ Stop end if
    Compute $w(\mu^{\ell+1}) \in W^{\mathscr{L}}$ as in (5.19)
    $S_1^{\ell+1} := S_1^\ell \cup \{\mu^{\ell+1}\}$, $\Xi_{\ell+1}^\lambda := \Xi_\ell^\lambda \cup \{w(\mu^{\ell+1})\}$
end for
return RB basis $\Xi_{N(\lambda)}^\lambda$
```

for a parameter μ_1^i chosen in the i-th step of Algorithm 1 with training set $\{\psi_\lambda\} \times \mathscr{D}_1^{\text{train}}$. We collect all such solutions of (5.19) in an *Evolution Library*

$$\mathbb{L}_{\text{evol}} := \{w_\lambda^1, \dots, w_\lambda^{N(\lambda)} \ : \lambda \in \Lambda, |\lambda| \leq J\},$$

which consists of all RB bases of the evolutions with initial values in $\mathbb{L}_{\text{init}}$.

We do not compute a reduced inf-sup stable test space $\mathbb{Y}_N$, since we use normal equations, see Remark 4 below. The space-time variational approach allows us to use a standard error estimator in Algorithm 1, i.e., the first part of (5.16),

$$\Delta_\ell^1(\mu) := \frac{\|r_{\ell,1}(\mu)\|_{Z'}}{\beta_{\text{LB}}}, \tag{5.20}$$

where β_{LB} is a lower bound of the inf-sup constant of the bilinear form b that may be determined a priori e.g. by eigenvalue computations.

Orthonormalization Note, that we are going to use combinations of RB bases stored in $\mathbb{L}_{\text{evol}}$ for the RB approximation online. These combinations of snapshots may not necessarily be linearly independent. We resolve this by performing an online orthonormalization that is prepared offline as follows. Denote the set of *all* functions that arise from the two-step Greedy method by $W^{N_{\max}} := \{w^i : 1 \leq i \leq N_{\max}\}$, $N_{\max} = \sum_{\lambda \in \Lambda, |\lambda| \leq J} N(\lambda)$, and denote its Gramian matrix by $\mathbf{M} := [(w^i, w^j)_{\mathbb{X}}]_{i,j=1,\dots,N_{\max}}$. We then compute an SVD, i.e., $\mathbf{M} = \mathbf{U}^T \mathbf{D} \mathbf{U}$. Setting $\mathbf{S} := \mathbf{U}^{-1} \mathbf{D}^{-1/2}$ obviously yields $\mathbf{S}^T \mathbf{M} \mathbf{S} = \mathbb{1}$. At a first glance, this might seem to be an overkill. However, in the online phase, we need to access all those rows and columns of $\mathbf{S}$ that correspond to the required RB snapshots in $\mathbb{L}_{\text{evol}}$ that are significant for approximating the evolution for a new μ_0. Hence, we need to consider $W^{N_{\max}}$.

Online Phase We need to adapt the online phase for the specific case of a parameter function μ_0. The main idea is to efficiently compute a (quasi-)best $N_0(\mu_0)$-term approximation to a given μ_0 by determining those $N_0(\mu_0)$ wavelets yielding the largest coefficients in absolute values. Let us denote by $\Lambda_N(\mu_0)$ the arising index set of dimension $N_0(\mu_0)$. Then, the computed approximation takes the form

$$u_{0,N}(\mu_0) = \sum_{\lambda \in \Lambda_N(\mu_0)} d_\lambda(\mu_0) q^\lambda, \qquad N_0(\mu_0) \equiv |\Lambda_N(\mu_0)|,$$

with $q^\lambda \in \mathbb{L}_{\text{init}}$. The approximation error can be estimated as

$$\begin{aligned} \|\mu_0 - u_{0,N}(\mu_0)\|_H \le \Delta_N^0(\mu_0) &:= \frac{\|r_{N,0}(\mu_0)\|_{H'}}{\beta_{\text{LB}}} \\ &\le \frac{1}{\beta_{\text{LB}}} \Big\| \mu_0 - \sum_{\lambda \in \Lambda_N(\mu_0)} d_\lambda \psi_\lambda \Big\|_H = \frac{1}{\beta_{\text{LB}}} \Big\| \sum_{\lambda \notin \Lambda_N(\mu_0)} d_\lambda \psi_\lambda \Big\|_H . \end{aligned} \tag{5.21}$$

For the evolutionary part, we use the reduced basis $\{w_\lambda^\ell : \lambda \in \Lambda_N(\mu), \ell = 1, \dots, N(\lambda)\} \subset \mathbb{L}_{\text{evol}}$ to span the reduced space, which is of dimension $N := N(\mu_0) = \sum_{\lambda \in \Lambda_N(\mu_0)} N_0(\lambda)$, i.e., we have to solve a linear system of dimension $N \times N$. Recall, that we have precomputed the SVD of the full Gramian matrix $\mathbf{M}$ along with the matrix $\mathbf{S}$ to orthogonalize the snapshots. Now, we pick the submatrix $\mathbf{S}_N$ of $\mathbf{S}$ consisting of the N rows and columns of $\mathbf{S}$ that correspond to the given initial value μ_0. We use this matrix $\mathbf{S}_N$ as a preconditioner for the system matrix $\mathbf{B}_N(\mu)$. The arising online phase is detailed in Algorithm 2.

Remark 4 The online solution in line 4 of Algorithm 2 is performed by solving the corresponding normal equations. This is also the reason why we did not construct a reduced test space in the offline phase in (5.19). For details, we refer to [4].

Recall from (5.16) that the residual—and consequently also the error estimate Δ_N—can be split in two parts. The first part contains $\|\mu_0 - u_{0,N}(\mu_0)\|_H$ which is equivalent to the initial value error in (5.21). Since $\mu_0 - u_{0,N}(\mu_0)$ is a linear combination of wavelets, the Riesz basis property allows us to reduce this computation to a weighted sum of wavelet coefficients, which is clearly online-efficient.

Algorithm 2 Online phase with online orthonormalization

```
Require: New parameter $\mu \in \mathcal{D}$, $N = N(\mu_0)$, $\mathbb{L}_{\text{init}}$, $\mathbb{L}_{\text{evol}}$, preconditioner $\mathbf{S}$.
1: Compute $\mathbf{B}_N(\mu)$, $\mathbf{F}_N(\mu)$ and pick $\mathbf{S}_N$ out of $\mathbf{S}$.
2: Orthogonalization: $\tilde{\mathbf{B}}_N(\mu) := \mathbf{S}_N^T \mathbf{B}_N(\mu) \mathbf{S}_N$, $\tilde{\mathbf{F}}_N(\mu) := \mathbf{S}_N^T \mathbf{F}_N(\mu)$.
3: if $\det(\tilde{\mathbf{B}}_N(\mu)) = 0$ then Delete zero rows/columns. end if
4: Solve $\tilde{\mathbf{u}}_N(\mu) = (\tilde{\mathbf{B}}_N(\mu))^{-1} \tilde{\mathbf{F}}_N(\mu) \to \mathbf{u}_N(\mu) = \mathbf{S}_N \tilde{\mathbf{u}}_N(\mu)$.
5: Compute $\Delta_N(\mu)$.
6: return RB solution $u_N(\mu)$, estimator $\Delta_N(\mu)$.
```

The second part of the error estimator is easily seen to be offline-online decomposable and thus computable online-efficient. In addition, we obtain the following error bound

$$\|g_1 - b_1(u_N(\mu), \cdot; \mu_1)\|_{\mathbb{Y}'} \leq \Big|1 - \sum_{\lambda \in \Lambda_N(\mu_0)} d_\lambda(\mu_0)\Big| \|g_1(\cdot)\|_{\mathbb{Y}'} + \Big|\sum_{\lambda \in \Lambda_N(\mu_0)} d_\lambda(\mu_0)\Big| \mathrm{tol}_1, \tag{5.22}$$

for tol_1 defined as in Algorithm 1 and $d_\lambda(\mu_0)$ being the expansion coefficients of the approximate initial condition in terms of the basis function q^λ of $\mathbb{L}_{\mathrm{init}}$ in (5.18).

It is remarkable that the upper bound (5.22) can be evaluated *before* the RB approximation is actually computed, i.e., a-priori. This is important since the RB solution may not respect the chosen greedy tolerances. The reason is that the training set of the parameter function space contains single functions but linear combination of these functions need to be considered online. If the upper bound (5.22) indicates this, one may need to add some more basis functions by performing some few offline computations (in a multi-fidelity fashion).

5.5 Numerical Experiments

We aim at numerically investigating the influence of the right-hand side and of the error made in the approximation of the parameter function onto the RB error (estimator). In order to concentrate on these issues, we consider a univariate diffusion problem for $V := H_0^1(0,1)$ and $H := L_2(0,1)$, $I := (0, 0.3)$

$$\begin{aligned} \dot{u}(t,x) - \mu_1 u''(t,x) &= g(x) && \text{for } (t,x) \in (0,0.3) \times (0,1), \\ u(0,x) &= \mu_0(x) && \text{for } x \in (0,1). \end{aligned}$$

The parameter space is chosen as $\mathscr{D} = \mathscr{D}_0 \times \mathscr{D}_1 := L_2(0,1) \times [0.5, 1.5]$. For the right-hand side, we compare two settings. The first one is $g(t,x) = g_{\mathrm{zero}}(t,x) \equiv 0$, as the a-priori error bound (5.22) is minimal then. The second case is an instationary smooth right-hand side $g(t,x) = g_{\sin}(t,x) = \sin(2\pi x)\cos(4\pi t)$.

Truth We use the space-time discretization that is equivalent to the Crank-Nicolson scheme, which is stable for $\Delta x = \Delta t = 2^{-6}$, [9]. Note, that this setting also allows us to compute the inf-sup constants analytically, so that our results are not influenced by any approximation errors in the constants. Finally, as in [9], we use the natural discrete norm for $w \in \mathbb{X}^{\mathscr{N}} \subset \mathbb{X}$ given as

$$|||w|||^2_{\mathscr{N}} := \|\bar{w}\|^2_{L_2(I;V)} + \|\dot{w}\|^2_{L_2(I;V')} + \|w(T)\|^2_H,$$

for $\bar{w}^k := (\Delta t)^{-1} \int_{I^k} w(t)\, dt \in V$ and $\bar{w} := \sum_{k=1}^{\mathscr{K}} \tau_k \otimes \bar{w}^k \in L_2(I;V)$.

Parameter Function For the representation of the initial value, we use the Haar wavelets, see Sect. 5.4.2. The approximation error, i.e., the sum of those wavelet coefficients that are not used in the approximation, is computed up to a sufficiently high fixed level. For the online computations, we used $\mu_{0,\text{smooth}} := x(1-x)$, which is smooth and allows for a sparse wavelet approximation. As a second example, we chose $\mu_{0,\text{L2}} := |x-0.5|^{1/2} \in L_2(0,1) \setminus H_0^1(0,1)$. The space-time formulation allows us such a non-smooth initial condition, whose wavelet coefficients reflect the singularity of the derivative at $x = 0.5$.

Evolution Greedy The tolerance was chosen as $\text{tol}_1 = 0.001$ and the training set for $\mathcal{D}_1$ was set $\mathcal{D}_1^{\text{train}} := \{0.5 + k\frac{1}{17} \ : \ k = 0, \dots, 17\}$, $|\mathcal{D}_1^{\text{train}}| = 18$. We fix the maximal level $J = 6$, which turned out to yield a sufficient resolution, i.e., $|\mathbb{L}_{\text{init}}| = 2^6 = 64$. The Evolution Greedy is performed $|S_J| = 2^6$ times with $\mathcal{D}^{\text{train}} = \{\psi_\lambda\} \times \mathcal{D}_1^{\text{train}}$ for all $|\lambda| < 6$. For both right-hand sides we obtain 4–5 evolution reduced bases functions. The results concerning the following quantities are displayed in Fig. 5.1.

(1)	$\|\|\|u(\mu) - u_{N_0+N}(\mu)\|\|\|_{\mathcal{N}}$	Exact error of the RB-approximation
(2)	$\Delta_N^1(\mu)$	the RB error bound for the evolution in (5.20)
(3)	$\frac{1}{\beta_{\text{LB}}}(\sum_{\lambda \notin \Lambda_N(\mu_0)} \|d_\lambda\|^2)^{1/2}$	sum of non-considered wavelet coefficients as upper bound for $\Delta_{N,0}(\mu_0)$ as in (5.21)
(4)	$\Delta_N^1(\mu) + \frac{1}{\beta_{\text{LB}}}(\sum_{\lambda \notin \Lambda_N(\mu_0)} \|d_\lambda\|^2)^{1/2}$	full error bound Δ_N: sum of the latter two
(5)	Bound	a-priori bound, right-hand side of (5.22)

In Fig. 5.1a (with smooth initial data and g_{zero}), we see almost no difference between (3) and (4), since the evolution error [estimator (2)] is very small, which should be expected for g_{zero}. Moreover, the difference between the full error estimator (4) and the true error (1) is quite small, the efficiency of the error estimator is quite good. The a-priori bound is reasonably good for $N_0 \geq 45$.

As we change the right-hand side to $g_{\sin}$ in Fig. 5.1b, the bound (5) immediately detects this. Until $N_0 = 50$, the error is dominated by the evolution and the error bound is quite sharp. However, as this part drops down, the initial value error (3) remains, which causes a slight decrease of efficiency.

The third case in Fig. 5.1c uses a non-smooth initial data. As expected, the decay of the initial value error (3) is slow—we need many wavelets to represent $\mu_{0,L2}$ well. The evolution error (2) is almost negligible, which is also detected by the a-priori bound (5). However, even in this case, the full error estimator is sharp. Of course, the final RB-dimension depends on the initial value and its approximation. However, we have compared several configurations of initial value and right-hand side and found in all cases that N grows linearly with N_0 and the ratio is the same in all scenarios [4].

In all cases, an expansion using $N_0 = 32$ wavelets already gives very good results. The detailed dimension was $\mathcal{N} = 4096$ and could be reduced to a maximum of $N = 160$, a factor of more than 25. We recall that the space-time approach

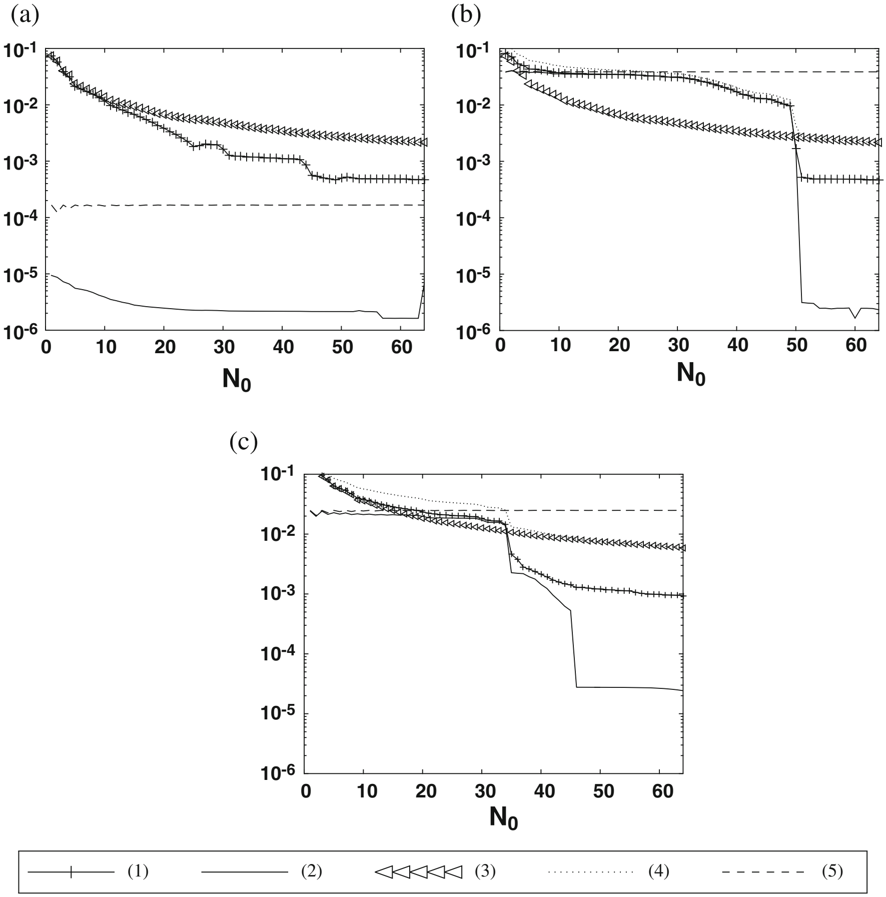


Fig. 5.1 Error and estimators for different cases. Quantities (1)–(5) according to table above. (**a**) μ_{smooth}; g_{zero}. (**b**) μ_{smooth}; $g_{\sin}$. (**c**) μ_{L2}; g_{zero}

amounts to solve one dense linear system of dimension N in the online stage. In order to determine the speedup (i.e., the comparison of offline and online cost), the best offline-situation occurs when a time-stepping scheme like Crank-Nicolson can be used, requiring in the order of $\frac{T}{\Delta t}$ solves of a sparse system. In our experiments, the corresponding CPU-time was about 0.01 s. The online RB-solution took about 0.0075 s, which is a speedup of 25%. Moreover, recall that our a-priori bound indicates the required RB-dimension in advance, so that the online cost can be estimated a priori.

However, such speedup numbers are misleading to a certain extend since they correspond to the best possible offline situation (time-stepping), which only occurs for very specific discretizations. Our approach also allows for non-time-marching truth discretizations in which case the speedup is by far larger (we refer to [4] for concrete applications in finance with an observed online speedup of about 97%).

We stress the fact that our numerical experiment is mainly designed to explain the functionality of the introduced method for parameter functions.

5.6 Conclusions

We presented a RBM for a parameter function as the initial value of a parabolic PDE. The space-time variational formulation allows us to separate the approximation of the initial condition from the error made in the evolution as time grows. An offline library for the initial value is suggested which guarantees a prescribed online tolerance. We used an online adaptive wavelet approximation, which provides us with a great flexibility regarding both the size of the RB spaces as well as the approximation quality. We present an a-priori bound as well as an error estimator.

The suggested offline library allows for arbitrary initial values in $L_2(0, 1)$ with prescribed accuracy. This flexibility results in a significantly large offline cost, in particular in higher space dimensions. To avoid this, one may use more specific approximation spaces for the initial value(s), which yields quite reasonable offline costs in particular in higher space dimensions [4]. Of course, this restricts flexibility and may result in a non-sufficient approximation of the initial value and consequently in a poor online reduced basis approximation.

In general, our numerical results show the flexibility of the method and the efficiency of the a posteriori error bound.

References

1. Ali, M., Urban, K.: Reduced basis exact error estimates with wavelets. In: Numerical Mathematics and Advanced Applications - ENUMATH 2015. Springer, Berlin (2016)
2. Ali, M., Steih, K., Urban, K.: Reduced basis methods based upon adaptive snapshot computations. Ulm University, preprint, arXiv:1407.1708, p. 30 (2014)
3. Haasdonk, B., Ohlberger, M.: Reduced basis method for finite volume approximations of parametrized linear evolution equations. Math. Model. Numer. Anal. **42**(2), 277–302 (2008)
4. Mayerhofer, A.: Reduced basis methods for parabolic PDEs with parameter functions in high dimensions and applications in finance. Ph.D. thesis, Ulm University (2016)
5. Mayerhofer, A., Urban, K.: A reduced basis method for parabolic partial differential equations with parameter functions and application to option pricing. J. Comput. Finance **20**(4), 71–106 (2017)
6. Schwab, C., Stevenson, R.: Space-time adaptive wavelet methods for parabolic evolution problems. Math. Comput. **78**(267), 1293–1318 (2009)
7. Showalter, R.E.: Monotone operators in Banach Space and Nonlinear Partial Differential Equations, vol. 49. American Mathematical Society, Providence (1997)
8. Urban, K.: Wavelet Methods for Elliptic Partial Differential Equations. Oxford University Press, Oxford (2009)
9. Urban, K., Patera, A.: A new error bound for reduced basis approximation of parabolic partial differential equations. C.R. Math. Acad. Sci. Paris **350**(3–4), 203–207 (2012)
10. Urban, K., Patera, A.: An improved error bound for reduced basis approximation of linear parabolic problems. Math. Comput. **83**(288), 1599–1615 (2014)

Chapter 6
Reduced Basis Isogeometric Mortar Approximations for Eigenvalue Problems in Vibroacoustics

Thomas Horger, Barbara Wohlmuth, and Linus Wunderlich

Abstract We simulate the vibration of a violin bridge in a multi-query context using reduced basis techniques. The mathematical model is based on an eigenvalue problem for the orthotropic linear elasticity equation. In addition to the nine material parameters, a geometrical thickness parameter is considered. This parameter enters as a 10th material parameter into the system by a mapping onto a parameter independent reference domain. The detailed simulation is carried out by isogeometric mortar methods. Weakly coupled patch-wise tensorial structured isogeometric elements are of special interest for complex geometries with piecewise smooth but curvilinear boundaries. To obtain locality in the detailed system, we use the saddle point approach and do not apply static condensation techniques. However within the reduced basis context, it is natural to eliminate the Lagrange multiplier and formulate a reduced eigenvalue problem for a symmetric positive definite matrix. The selection of the snapshots is controlled by a multi-query greedy strategy taking into account an error indicator allowing for multiple eigenvalues.

6.1 Introduction

Eigenvalue problems in the context of vibroacoustics often depend on several parameters. In this work, we consider a geometry and material dependent violin bridge. For a fast and reliable evaluation in the real-time and multi-query context, reduced basis methods have proven to be a powerful tool.

For a comprehensive review on reduced basis methods, see, e. g., [29, 33] or [28, Chap. 19] and the references therein. The methodology has been applied successfully to many different problem classes, among others Stokes problems [20, 22, 32, 34], variational inequalities [13, 15] and linear elasticity [24]. Recently, reduced basis methods for parameterized elliptic eigenvalue problems (μEVPs)

T. Horger (✉) • B. Wohlmuth • L. Wunderlich
Institute for Numerical Mathematics, Technische Universität München, Boltzmannstraße 3, 85748 Garching b. München, Germany
e-mail: horger@ma.tum.de; wohlmuth@ma.tum.de; linus.wunderlich@ma.tum.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_6

gained attention. Early work on a residual based a posteriori estimator for the first eigenvalue can be found in [23] and has been generalized in [26, 27] to the case of several single eigenvalues with special focus to applications in electronic structure problems in solids. Furthermore, the very simple and special case of a single eigenvalue where only the mass matrix and not the stiffness matrix of a generalized eigenvalue problem is parameter dependent has been discussed in [11]. Alternatively to the classical reduced basis approach, component based reduction strategies are considered in [36]. Here, we follow the ideas of [17] where rigorous bounds in the case of multi-query and multiple eigenvalues are given. More precisely, a single reduced basis is built for all eigenvalues of interest. The construction is based on a greedy strategy using an error estimator which can be decomposed into offline and online components.

The eigenvalues of a violin bridge play a crucial role in transmitting the vibration of the strings to the violin body and hence influence the sound of the instrument, see [10, 37]. Due to the complicated curved domain and improved eigenvalue approximations compared to finite element methods, see [19], we consider an isogeometric discretization. Flexibility for the tensor product spline spaces are gained by a weak domain decomposition of the non-convex domain.

Isogeometric analysis, introduced in 2005 by Hughes et al. in [18], is a family of methods that uses B-splines and non-uniform rational B-splines (NURBS) as basis functions to construct numerical approximations of partial differential equations, see also [1, 5]. Mortar methods are a popular tool for the weak coupling of non-matching meshes, originally introduced for spectral and finite element methods [2, 3]. An early contribution to isogeometric elements in combination with domain decomposition techniques can be found in [16]. A rigorous mathematical analysis of uniform inf-sup stability and reproduction properties for different Lagrange multiplier spaces is given in [4]. Applications of isogeometric mortar methods can be found in [7, 8, 35]. The weak form of a discrete mortar approach can be either stated as a positive definite system on a constrained primal space or alternatively as an indefinite saddle point system in terms of a primal and dual variable. Both formulations are equivalent in the sense that they do yield the same primal solution. From the computational point of view, quite often the saddle point formulation is preferred since it allows the use of locally defined basis functions and yields sparse systems. The elimination of the dual variable involves the inverse of a mass matrix and, unless biorthogonal basis functions are used, significantly reduces the sparsity pattern of the stiffness matrix. In general, the constrained primal basis functions have a global support on the slave side of the interface. This observation motivates us to use for the computation of the detailed solution the saddle point mortar formulation and work with locally defined unconstrained basis functions yielding sparse systems. However, typically a reduced system is automatically dense. If the constraint is parameter independent we obtain a positive definite system for the reduced setting. Here we show that even in the situation of a parameter dependent geometry, we can reformulate the weak continuity constraint in a parameter independent way.

The rest of this contribution is structured as follows. In Sect. 6.2, we introduce the model problem and briefly discuss the assumed orthotropic material law and the applied isogeometric mortar discretization for the violin bridge. The geometric setup includes a thickness parameter which is transformed to a material parameter. Here, we also comment on the fact that although the geometry transformation formally brings in a parameter into the weak mortar coupling, we can recast the problem formulation as a parameter independent coupling condition across the interfaces. The reduced basis approach is given in Sect. 6.3. Finally numerical results illustrating the accuracy and flexibility of the presented approach are given in Sect. 6.4. We point out that our parameter space is possibly non-convex due to the non-linear constraints of the material parameters.

6.2 Problem Setting

The numerical simulation of vibroacoustic applications involves quite often complex domains. Typical examples are large structures, such as, e.g., bridges, technical devices such as, e.g., loudspeakers but also parts of string instruments such as, e.g. violin bridges see Fig. 6.1. Within the abstract framework of modal analysis, the fully bi-directional mechanical-acoustic coupled system can be reduced to a generalized eigenvalue problem.

For the three dimensional geometry of a violin bridge, we consider the eigenvalue problem of elasticity

$$-\operatorname{div}\sigma(u) = \lambda\rho u,$$

where $\rho > 0$ is the mass density, and $\sigma(u)$ depends on the material law of the structure under consideration. In our case, linear orthotropic materials are appropriate since as depicted in Fig. 6.2 wood consists of three different axes and only small deformations are considered. Note that besides the cylindrical structure of a tree trunk, we consider Cartesian coordinates due to the small size of the violin bridge compared to the diameter of a tree trunk.

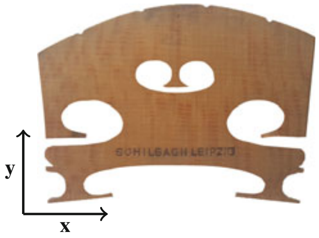


Fig. 6.1 Example of a violin bridge

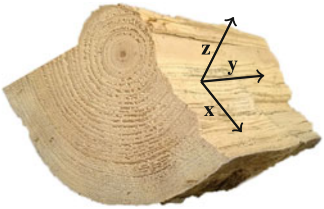


Fig. 6.2 Illustration of the orthotropic structure of wood

6.2.1 *Orthotropic Material Law*

The three axes are given by the fiber direction y, the in plane orthogonal direction z and the radial direction x. By Hooke's law, the stress strain relation can be stated in its usual form as $\sigma(u) = \mathbb{C}\varepsilon(u)$ with $\varepsilon(u) = (\nabla u + \nabla u^\top)/2$. Due to the alignment of the coordinate system with the orthotropic structure, the stiffness tensor is given as

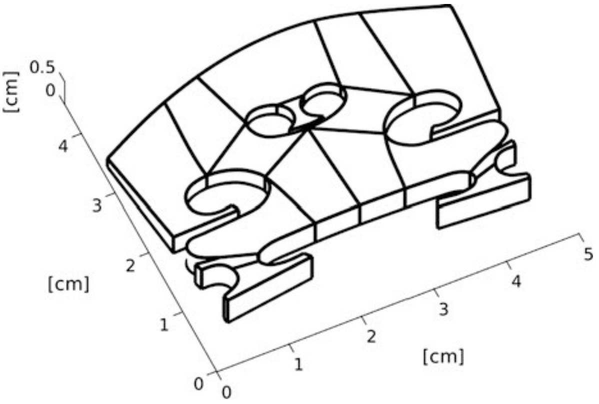
$$\mathbb{C} = \begin{pmatrix} A_{11} & A_{12} & A_{13} & 0 & 0 & 0 \\ A_{21} & A_{22} & A_{23} & 0 & 0 & 0 \\ A_{31} & A_{32} & A_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & G_{yz} & 0 & 0 \\ 0 & 0 & 0 & 0 & G_{zx} & 0 \\ 0 & 0 & 0 & 0 & 0 & G_{xy} \end{pmatrix}, \tag{6.1}$$

with the shear moduli G_{xy}, G_{yz}, G_{zx} and the entries A_{ij} depending on the elastic moduli E_x, E_y, E_z and the Poisson's ratios ν_{xy}, ν_{yz}, ν_{zx}. The exact formula for A_{ij} can be found in [30, Chap. 2.4].

Some important differences compared to isotropic material laws are worth pointing out. While in the isotropic case, all Poisson's ratios share the same value, for orthotropic materials they represent three independent material parameters. The only relation between the ratios is $\nu_{ij}E_j = \nu_{ji}E_i$. Also the possible range of the material parameters, i.e., $-1 < \nu < 1/2$ for the isotropic case, is different. A positive definite stiffness tensor and thus a coercive energy functional is only guaranteed if $1 > \nu_{yz}^2 E_z/E_y + \nu_{xy}^2 E_y/E_x + 2\nu_{xy}\nu_{yz}\nu_{zx}E_z/E_x + \nu_{zx}^2 E_z/E_x$ and $E_x/E_y > \nu_{xy}^2$. Note that Poisson's ratios larger than $1/2$ are permitted, but this does not imply unphysical behavior as in the isotropic case, see, e.g., [31]. The conditions $E_i, G_{ij} > 0$ hold both in the isotropic and orthotropic case.

The curved domain of the violin bridge can be very precisely described by a spline volume. Since it is not suitable for a single-patch description, we decompose it into 16 three-dimensional spline patches shown in Fig. 6.3. While the description of the geometry could also be done with fewer patches, the number of 16 patches Ω_l gives us regular geometry mappings and a reasonable flexibility of the individual meshes.

Fig. 6.3 Decomposition of the three-dimensional geometry into 16 patches Ω_l and 16 interfaces γ_k



The decomposed geometry is solved using an equal-order isogeometric mortar method as described in [4]. A trivariate B-spline space V_l is considered on each patch Ω_l. The broken ansatz space $V_h = \prod_l V_l$ is weakly coupled on each of the 16 interfaces. For each interface γ_k the two adjacent domains are labeled as one slave and one master domain (i.e. $\gamma_k = \partial\Omega_{s(k)} \cap \partial\Omega_{m(k)}$) and the coupling space M_k is set as a reduced trace space of the spline spaces on the slave domain, i.e., $M_h = \prod_k M_k$. Several crosspoints and wirebasket lines exist in the decomposition where an appropriate local degree reduction has to be performed to guarantee uniform stability, see [4, Sect. 4.3].

We use the standard bilinear forms for mortar techniques in linear elasticity

$$a(u,v) = \sum_l \int_{\Omega_l} \sigma(u) : \varepsilon(v), \quad m(u,v) = \sum_l \int_{\Omega_l} \rho u v, \quad b(v,\widehat{\tau}) = \sum_k \int_{\gamma_k} [v]_k \widehat{\tau},$$

where $[v]_k = v_{s(k)}\big|_{\gamma_k} - v_{m(k)}\big|_{\gamma_k}$ denotes the jump across the interface γ_k. We note that no additional variational crime by different non-matching geometrical resolutions of γ_k enters. The detailed eigenvalue problem is defined as $(u,\tau) \in V_h \times M_h$, $\lambda \in \mathbb{R}$, such that

$$a(u,v) + b(v,\tau) = \lambda m(u,v), \qquad v \in V_h, \tag{6.2a}$$

$$b(u,\widehat{\tau}) = 0, \qquad \widehat{\tau} \in M_h. \tag{6.2b}$$

We note that the constraint (6.2b) reflects the weak continuity condition of the displacement across the interface with respect to the standard two-dimensional Lebesgue measure. Only in very special situations strong point-wise continuity is granted from (6.2b).

6.2.2 Transforming Geometrical Parameters to Material Parameters

Additional to the nine material parameters E_i, G_{ij}, ν_{ij}, we consider a geometry parameter μ_{10} related to the thickness of the violin bridge. Transforming the geometry to a reference domain, we can interpret the thickness parameter as extra material parameter. Let the parameter dependent geometry $\Omega(\mu)$ be a uni-directional scaling of a reference domain $\widehat{\Omega}$, i.e., a transformation by $F(\cdot;\mu) : \widehat{\Omega} \to \Omega(\mu)$, $\mathbf{x} = F(\widehat{\mathbf{x}};\mu) = (\widehat{x}, \widehat{y}, \mu_{10}\widehat{z})$, with $\widehat{\mathbf{x}} = (\widehat{x}, \widehat{y}, \widehat{z}) \in \widehat{\Omega}$. Transforming the unknown displacement and rescaling it as $\widehat{u}(\widehat{\mathbf{x}}) = DF(\widehat{\mathbf{x}};\mu)^\top u(F(\widehat{\mathbf{x}};\mu))$ allows us to define a symmetric strain variable on the reference domain

$$\widehat{\varepsilon}(\widehat{u}(\widehat{\mathbf{x}})) = DF(\widehat{\mathbf{x}};\mu)^\top \varepsilon(u(F(\widehat{\mathbf{x}};\mu)))DF(\widehat{\mathbf{x}};\mu).$$

The orthotropic stiffness tensor (6.1) is then transformed to

$$\widehat{\mathbb{C}}(\mu) = \begin{pmatrix} A_{11} & A_{12} & \mu_{10}^{-2}A_{13} & & & \\ A_{21} & A_{22} & \mu_{10}^{-2}A_{23} & & & \\ \mu_{10}^{-2}A_{31} & \mu_{10}^{-2}A_{32} & \mu_{10}^{-4}A_{33} & & & \\ & & & \mu_{10}^{-2}G_{yz} & & \\ & & & & \mu_{10}^{-2}G_{zx} & \\ & & & & & G_{xy} \end{pmatrix}.$$

In terms of this coordinate transformation, the eigenvalue problem in the continuous H^1-setting reads, since $\det DF(\widehat{\mathbf{x}};\mu) = \mu_{10}^{-1}$ is constant, as

$$\int_{\widehat{\Omega}} \widehat{\varepsilon}(\widehat{u})\, \widehat{\mathbb{C}}(\mu)\, \widehat{\varepsilon}(\widehat{v})\, \mathrm{d}\widehat{\mathbf{x}} = \lambda \int_{\widehat{\Omega}} \rho\, \widehat{u}^\top \begin{pmatrix} 1 & & \\ & 1 & \\ & & \mu_{10}^{-2} \end{pmatrix} \widehat{v}\, \mathrm{d}\widehat{\mathbf{x}}.$$

In the mortar case, the coupling conditions across the interfaces have to be transformed as well. Here we assume that the meshes on the physical domain are obtained from the same mesh on the reference domain by the mapping F.

$$\begin{aligned} \int_{\gamma(\mu)} [u(\mathbf{x})]\tau(\mathbf{x})\, \mathrm{d}\gamma(\mathbf{x}) &= \int_{\widehat{\gamma}} [DF(\widehat{\mathbf{x}};\mu)^{-\top}\widehat{u}(\widehat{\mathbf{x}})]\tau(F(\widehat{\mathbf{x}};\mu))\mu_{10}\, \mathrm{d}\widehat{\gamma}(\widehat{\mathbf{x}}) \\ &= \mu_{10} \int_{\widehat{\gamma}} [\widehat{u}(\widehat{\mathbf{x}})] \begin{pmatrix} 1 & & \\ & 1 & \\ & & \mu_{10}^{-1} \end{pmatrix} \tau(F(\widehat{\mathbf{x}};\mu))\, \mathrm{d}\widehat{\gamma}(\widehat{\mathbf{x}}). \end{aligned}$$

We note that $\widehat{\tau} := \tau \circ F$ is in the parameter independent Lagrange multiplier space on the reference domain if τ is in the parameter dependent one on the physical domain. The remaining parameter dependence is a pure scaling of the Lagrange multiplier, which does not influence the constrained primal space. These considerations show us that the standard mortar coupling which is due to the geometry variation parameter dependent can be transformed to a parameter independent one.

While these lines use the special structure of the geometry variation F, the coupling can be transformed to a parameter independent one even in more general situations. Then, the coupling on $\Omega(\mu)$ must be posed in a suitable weighted L^2-space, which is adapted such that the transformed coupling is parameter independent.

Another material parameter is the constant mass density ρ. However it does not influence the eigenvectors. Only the eigenvalue is rescaled, yielding a trivial parameter dependence. For this reason, the density is kept constant in the reduced basis computations and can be varied in a post-process by rescaling the eigenvalues.

The described material parameters allow for an affine parameter dependence of the mass and the stiffness, with $Q_a = 10$, $Q_m = 2$,

$$a(\cdot,\cdot;\mu) = \sum_{q=1}^{Q_a} \theta_a^q(\mu) a^q(\cdot,\cdot), \quad m(\cdot,\cdot;\mu) = \sum_{q=1}^{Q_m} \theta_m^q(\mu) m^q(\cdot,\cdot),$$

where $\theta_m^1(\mu) = 1$ can be chosen parameter independent.

6.3 Reduced Basis

We now apply reduced basis (RB) methods for the approximation of the parameter dependent eigenvalue problem on the reference domain. By abuse of notation, we denote the spaces and bilinear forms transformed to the reference domain as before. RB techniques where the detailed problem is in saddle point form, in general, require the construction of RB for both the primal and the dual space, see, e.g., variational inequalities or when the coupling is parameter dependent, see [12, 14, 15, 25]. To ensure the inf-sup stability of the discrete saddle point problem, supremizers can be added to the primal space, additionally increasing the size of the reduced system, see [32, 34]. Here it is sufficient to define a RB for the primal space. For the simultaneous approximation of possible multiple eigenvalues and eigenvectors, we follow the approach given and analyzed in [17].

Due to the parameter independence of $b(\cdot,\cdot)$ and the dual space, obtained by the transformation described above, we can reformulate the detailed saddle point problem (6.2) in a purely primal form posed on the constrained space

$$X_h = \{v \in V_h, b(v,\widehat{\tau}) = 0, \widehat{\tau} \in M_h\}.$$

We recall that this formulation is not suitable for solving the detailed solution, since, in general, it is costly to construct explicitly a basis of X_h and severely disturbs the sparsity of the detailed matrices.

The construction of the RB functions is done in two steps. Firstly, an initial basis is built by a small POD from detailed solutions. This basis is then enlarged by a greedy algorithm using an asymptotically reliable error estimator. All detailed solutions do satisfy the weak coupling property and by definition the RB functions do as well. Thus the saddle-point problem is reduced to a positive definite one. The eigenvalue problem on the reduced space $X_N = \{\zeta_n \in X_h, n = 1, \dots, N\}$, for the first K eigenpairs is then given by: Find the eigenvalues $\lambda_{\mathrm{red},\,i}(\mu) \in \mathbb{R}$ and the eigenfunctions $u_{\mathrm{red},\,i}(\mu) \in X_N$, $i = 1, \dots, K$, such that

$$a(u_{\mathrm{red},\,i}(\mu), v; \mu) = \lambda_{\mathrm{red},\,i}(\mu)\, m(u_{\mathrm{red},\,i}(\mu), v; \mu), \qquad v \in X_N.$$

The error estimator presented in [17, Corollary 3.3] can directly be applied, but the online-offline decomposition needs to be modified. In the original setting, a parameter independent mass was considered, so we need to additionally include the affine decomposition of the mass matrix.

The definition of the estimator is based on the residual

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

measured in the dual norm $\|\cdot\|_{\hat{\mu};X_h'}$, with $\|g\|_{\hat{\mu};X_h'} = \sup_{v \in X_h} g(v)/\hat{a}(v,v)^{1/2}$ for $g \in X_h'$, where $\hat{a}(u,v) := a(u,v;\widehat{\mu})$, and $\widehat{\mu} \in \mathscr{P}$ is a reference parameter. We define $\hat{e}_i(\mu) \in X_h$ by

$$\hat{a}(\hat{e}_i(\mu), v) = r_i(v; \mu), \qquad v \in X_h.$$

To adapt the online-offline decomposition, we follow [17, 23] and add additional terms corresponding to the mass components $m^q(\cdot,\cdot)$. The decomposition of the mass can be related to the already known decomposition of the stiffness matrix, by formally defining a bilinear form $a(u,v;\mu) - \lambda_{\mathrm{red},\,i}(\mu)\, m(u,v;\mu)$. For the convenience of the reader we recall the main steps. Let $(\zeta_n)_{1 \le n \le N}$ be a orthonormal basis (w. r. t. $m(\cdot,\cdot;\widehat{\mu})$) of X_N and let us define $\xi_n^q \in X_N$ and $\xi_n^{m,q} \in X_N$ by

$$\begin{aligned} \hat{a}(\xi_n^q, v) &= a^q(\zeta_n, v), \qquad v \in X_h,\ 1 \le n \le N,\ 1 \le q \le Q_a, \\ \hat{a}(\xi_n^{m,q}, v) &= m^q(\zeta_n, v), \qquad v \in X_h,\ 1 \le n \le N,\ 1 \le q \le Q_m. \end{aligned}$$

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

Then, given a reduced eigenpair $(u_{\mathrm{red},i}(\mu), \lambda_{\mathrm{red},i}(\mu))$, we have the error representation

$$\hat{e}_i(\mu) = \sum_{n=1}^{N}\sum_{q=1}^{Q_a} \theta_a^q(\mu)\,(u_{\mathrm{red},i}(\mu))_n\,\xi_n^q - \lambda_{\mathrm{red},i}(\mu)\sum_{n=1}^{N}\sum_{q=1}^{Q_m}\theta_m^q(\mu)\,(u_{\mathrm{red},i}(\mu))_n\,\xi_n^{m,q}.$$

Consequently, using $\|r_i(\cdot;\mu)\|^2_{\hat{\mu};X_h'} = \hat{a}(\hat{e}_i(\mu), \hat{e}_i(\mu))$, the computational cost intense part of the error estimator can be performed in the offline phase, see [17, Sect. 3.3] for a more detailed discussion.

6.4 Numerical Simulation

In this section, the performance of the proposed algorithm is illustrated by numerical examples. The detailed computations were performed using geoPDEs [6], a Matlab toolbox for isogeometric analysis, the reduced computations are based on RBmatlab [9].

For the detailed problem, we use an anisotropic discretization. In plane, we use splines of degree $p = 3$ on the non-matching mesh shown in Fig. 6.4. The mesh has been adapted locally to better resolve possible corner singularities of the solution. In the z-direction a single element of degree $p = 4$ is used. The resulting equation system has 45,960 degrees of freedom for the displacement whereas the surface traction on the interfaces is approximated by 2025 degrees of freedom.

We consider the ten parameters, described in Sect. 6.2, $\mu = (\mu_1, \ldots, \mu_{10})$ with the elastic modulii $\mu_1 = E_x$, $\mu_2 = E_y$, $\mu_3 = E_z$, the shear modulii $\mu_4 = G_{yz}$, $\mu_5 = G_{xz}$, $\mu_6 = G_{xy}$, Poisson's ratios $\mu_7 = \nu_{yz}$, $\mu_8 = \nu_{xz}$, $\mu_9 = \nu_{xy}$ and the scaling of the thickness μ_{10}.

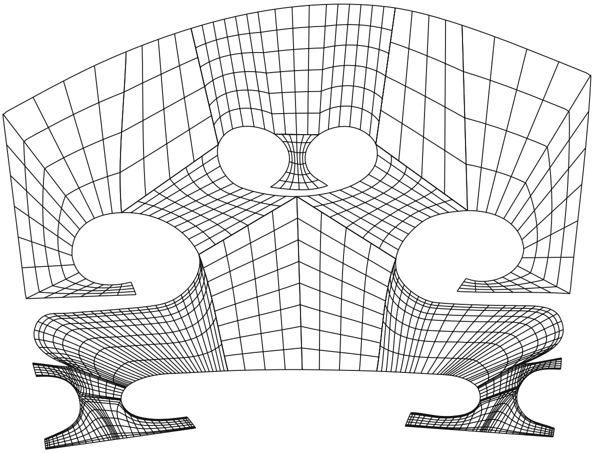


Fig. 6.4 Non-matching isogeometric mesh of the violin bridge

The considered parameter values were chosen according to real parameter data given in [31, Table 7-1]. We consider two different scenarios. In the first setting, we fix the wood type and take into account only natural variations, see [31, Sect. 7.10]. To capture the sensitivity of the violin bridge with respect to uncertainty in the material parameter one can chose a rather small parameter range around the reference parameter. We chose the reference data of *Fagus sylvatica*, the common beech, as given in Table 6.1, as well as the parameter range $\mathscr{P}_1$. The mass density is fixed in all cases as 720 kg/m^3.

In our second test setting, we also consider different wood types. Hence we have to consider a larger parameter set, including the parameters for several types of wood, resulting in a larger parameter set $\mathscr{P}_2$, see Table 6.1. We note, that not all parameters in this large range are admissible for the orthotropic elasticity as they do not fulfill the conditions for the positive definiteness of the elastic tensor, stated in Sect. 6.2.1. Thus, we constrain the tensorial parameter space by

$$1 - \nu_{yz}^2 E_z/E_y + \nu_{xy}^2 E_y/E_x + 2\nu_{xy}\nu_{yz}\nu_{zx}E_z/E_x + \nu_{zx}^2 E_z/E_x \geq c_0,$$

as well as $E_x/E_y - \nu_{xy}^2 \geq c_1$ where the tolerances $c_0 = 0.01$ and $c_1 = 0.01$ were chosen, such that the wood types given in [31, Sect. 7.10] satisfy these conditions. Exemplary, in Fig. 6.5 we depict an lower-dimensional sub-manifold of $\mathscr{P}_2$ which includes non-admissible parameter values.

Table 6.1 Reference parameter and considered parameter ranges

	E_x [MPa]	E_y [MPa]	E_z [MPa]	G_{yz} [MPa]	G_{zx} [MPa]	G_{xy} [MPa]	ν_{yz}	ν_{zx}	ν_{xy}
$\widehat{\mu}$	14,000	2280	1160	465	1080	1640	0.36	0.0429	0.448
$\mathscr{P}_1$	13,000	1500	750	100	500	1000	0.3	0.03	0.4
	−15,000	−3000	−1500	−1000	−1500	−2000	−0.4	−0.06	−0.5
$\mathscr{P}_2$	1000	100	100	10	100	100	0.1	0.01	0.3
	−20,000	−5000	−2000	−5000	−2500	−5000	−0.5	−0.1	−0.5

Fig. 6.5 Illustration of non-admissible parameter values in a lower-dimensional sub-manifold of $\mathscr{P}_2$, varying $\nu_{zx} \in (0.01, 0.1)$, $\nu_{xy} \in (0.3, 0.5)$, $E_y \in (100, 5000)$ and fixing $E_x = 1000$, $E_z = 2000$ and $\nu_{yz} = 0.5$

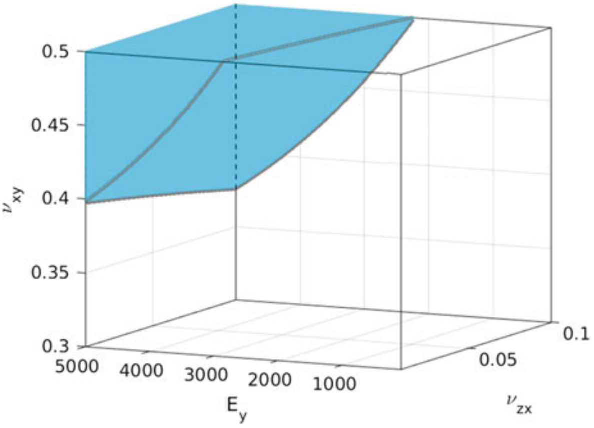


Table 6.2 The ten smallest eigenvalues for different thickness parameters, with the other parameters fixed to the reference value

Eigenvalue	$\mu_{10} = 0.5$	$\mu_{10} = 1.0$	$\mu_{10} = 2.0$	Ratio 0.5/1.0	Ratio 1.0/2.0
1	0.4057	1.3238	3.6954	0.3065	0.3582
2	1.1613	3.8870	10.8071	0.2988	0.3597
3	4.4096	12.9562	26.5621	0.3403	0.4878
4	6.1371	19.3254	30.0050	0.3176	0.6441
5	13.5564	27.3642	53.2657	0.4954	0.5137
6	19.2229	46.2521	93.9939	0.4156	0.4921
7	27.6118	65.0940	111.6075	0.4242	0.5832
8	39.3674	96.8069	129.3406	0.4067	0.7485
9	57.8266	107.6749	189.6090	0.5370	0.5679
10	68.0131	130.8876	241.7695	0.5196	0.5414

The thickness parameters is chosen to vary between 1/2 and 2 with the reference value set to 1

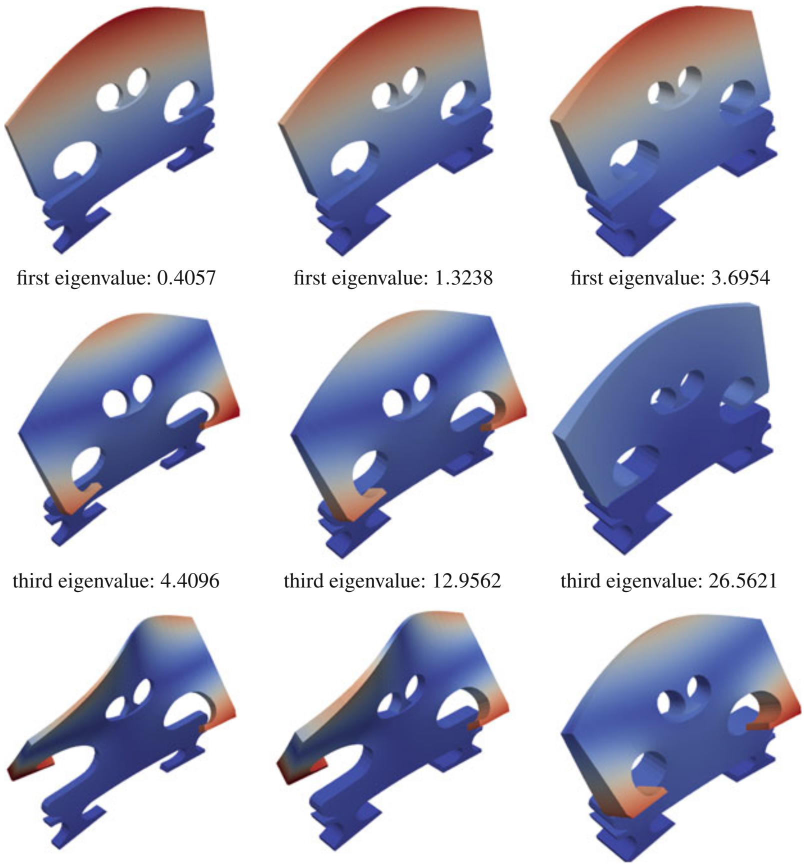
First, we consider the effect of the varying thickness parameter on the solution of our model problem. In Table 6.2 the first eigenvalues are listed for different values of the thickness, where we observe a notable and nonlinear parameter dependency. A selection of the corresponding eigenfunctions is depicted in Fig. 6.6, where the strong influence becomes even more evident, since in some cases the shape of the eigenmode changes when varying the thickness.

In the following RB tests, the relative error values are computed as the mean value over a large amount of random parameters. The L^2-error of the normed eigenfunctions is evaluated as the residual of the L^2-projection onto the corresponding detailed eigenspace. This takes into account possible multiple eigenvalues and the invariance with respect to a scaling by (-1).

The first test is the simultaneous approximation of the first five eigenpairs on both parameter sets $\mathscr{P}_1$ and $\mathscr{P}_2$. We use an initial basis of size 25 computed by a POD, which is enriched by the greedy algorithm up to a basis size of 250. In Fig. 6.7, the error decay for the different eigenvalues and eigenfunctions is presented. We observe very good convergence, with a similar rate in all cases. As expected the magnitude of the error grows with the dimension and range of the parameter set.

At this point, for the sake of completeness, we also consider the effectivities of the error estimator and the resulting speed-up. For example using the parameter range $\mathscr{P}_1$ varying the thickness, effectivities are around 4–16. Using the largest RB of dimension 250, the computational speedup of the eigenvalue solver in Matlab is a factor of 552.

Also an approximation of a larger number of eigenpairs does not pose any unexpected difficulties. Error values for the eigenvalue and eigenfunction are shown



first eigenvalue: 0.4057 first eigenvalue: 1.3238 first eigenvalue: 3.6954

third eigenvalue: 4.4096 third eigenvalue: 12.9562 third eigenvalue: 26.5621

fourth eigenvalue: 6.1371 fourth eigenvalue: 19.3254 fourth eigenvalue: 30.0050

Fig. 6.6 Influence of the thickness of the bridge on several eigenfunctions

in Fig. 6.8 for an approximation of the first 15 eigenpairs in the parameter set $\mathscr{P}_1$, showing a good convergence behavior. The RB size necessary for a given accuracy increases compared to the previous cases of 5 eigenpairs, due to the higher amount of eigenfunctions which are, for a fixed parameter, orthogonal to each other.

When considering the relative error for the eigenvalues, see Figs. 6.7 and 6.8, we note that for a fixed basis size, the higher eigenvalues have a better relative approximation than the lower ones. In contrast, considering the eigenfunctions, the error of the ones associated with the lower eigenvalues are smaller compared to the

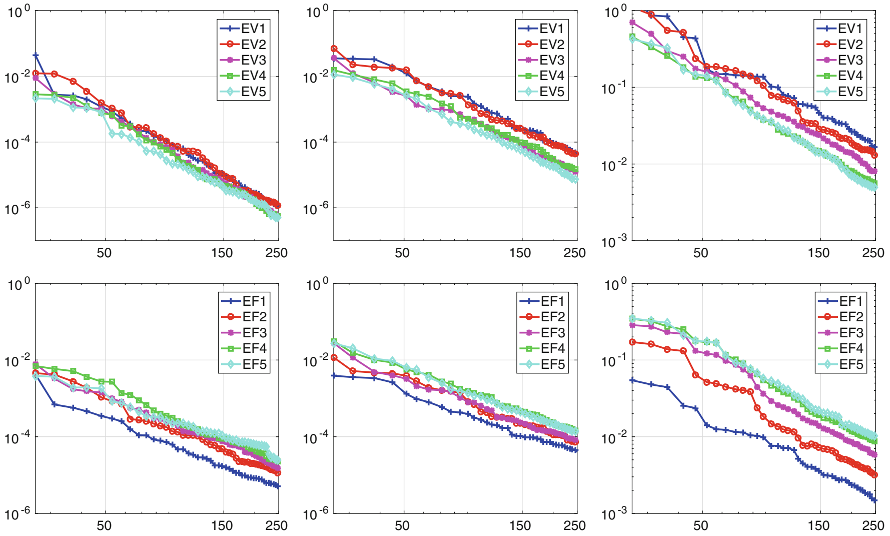


Fig. 6.7 Convergence of the relative error of the eigenvalues (*top*) and eigenfunctions (*bottom*). Parameter range $\mathscr{P}_1$ with a fixed thickness (*left*), with varying thickness (*middle*) and parameter range $\mathscr{P}_2$ with varying thickness (*right*)

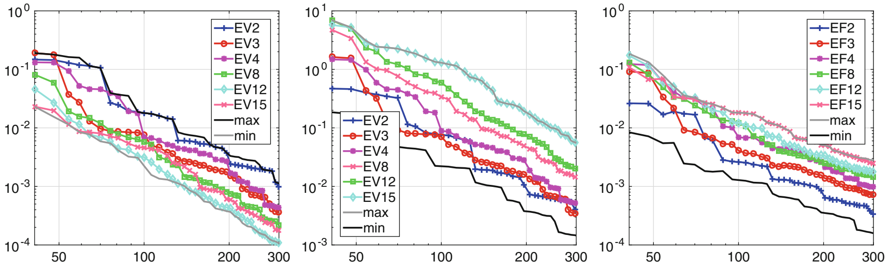
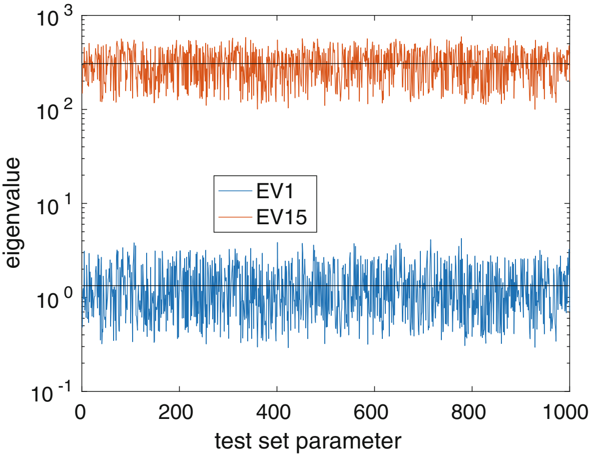


Fig. 6.8 Convergence of the relative (*left*), absolute (*middle*) error of the eigenvalues and eigenfunctions (*right*). Parameter range $\mathscr{P}_1$ with varying thickness, simultaneous approximating 15 eigenpairs

ones associated with the higher eigenvalues. This observation also holds true for the absolute error in the eigenvalues. This is related to the fact that eigenvalues depend sensitively on the parameters. In Fig. 6.9, we illustrate the distribution of the first and 15th eigenvalue.

Fig. 6.9 Sampling of the first and 15th eigenvalue within the parameter set $\mathscr{P}_1$ as used in the test set. Extremal values: $\min \lambda_1 = 0.29$, $\max \lambda_1 = 4.24$, $\min \lambda_{15} = 100.19$, $\max \lambda_{15} = 593.65$



6.5 Conclusion

We have considered generalized eigenvalue problems to approximate the vibrations of parameter dependent violin bridges. The model reduction is carried out in terms of a RB method where the detailed solutions are obtained by isogeometric mortar finite elements. In all considered test scenarios, highly accurate approximations for both eigenvalues and eigenmodes are obtained. At the same time the complexity and thus the run-time is significantly reduced. Instead of a detailed saddle point system with 47,985 degrees of freedom, we have only to solve eigenvalue problems on positive-definite systems with less than 300 degrees of freedom. Of special interest is not only the variation in the material parameter but also to take into account possible changes in the thickness of the violin bridge. In terms of a mapping to a reference domain, we can reinterpret the geometry parameter as an additional material parameter and avoid the indefinite saddlepoint problem.

Acknowledgements We would like to gratefully acknowledge the funds provided by the “Deutsche Forschungsgemeinschaft” under the contract/grant numbers: WO 671/11-1, WO 671/13-2 and WO 671/15-1 (within the Priority Programme SPP 1748, “Reliable Simulation Techniques in Solid Mechanics. Development of Non-standard Discretisation Methods, Mechanical and Mathematical Analysis”).

References

1. Beirão Da Veiga, L., Buffa, A., Sangalli, G., Vázquez, R.: Mathematical analysis of variational isogeometric methods. Acta Numer. **23**, 157–287 (2014)
2. Ben Belgacem, F., Maday, Y.: The mortar finite element method for three dimensional finite elements. Math. Model. Numer. Anal. **31**(2), 289–302 (1997)

3. Bernardi, C., Maday, Y., Patera, A.T.: A new nonconforming approach to domain decomposition: the mortar element method. In: Brezis, H. et al. (eds.) Nonlinear Partial Differential Equations and Their Applications, vol. XI, pp. 13–51. Collège de France, Paris (1994)
4. Brivadis, E., Buffa, A., Wohlmuth, B., Wunderlich, L.: Isogeometric mortar methods. Comput. Methods Appl. Mech. Eng. **284**, 292–319 (2015)
5. Cottrell, J.A., Hughes, T.J.R., Bazilevs, Y.: Isogeometric analysis. Towards Integration of CAD and FEA. Wiley, Chichester (2009)
6. de Falco, C., Reali, A., Vázquez, R.: GeoPDEs: a research tool for isogeometric analysis of PDEs. Adv. Eng. Softw. **42**(12), 1020–1034 (2011)
7. Dittmann, M., Franke, M., Temizer, I., Hesch, C.: Isogeometric analysis and thermomechanical mortar contact problems. Comput. Methods Appl. Mech. Eng. **274**, 192–212 (2014)
8. Dornisch, W., Vitucci, G., Klinkel, S.: The weak substitution method – an application of the mortar method for patch coupling in NURBS-based isogeometric analysis. Int. J. Numer. Methods Eng. **103**(3), 205–234 (2015)
9. Drohmann, M., Haasdonk, B., Kaulmann, S., Ohlberger, M.: A software framework for reduced basis methods using DUNE -RB and RBmatlab. Advances in DUNE, pp. 77–88. Springer, Berlin (2012)
10. Fletcher, N.H., Rossing, T.: The Physics of Musical Instruments, 2nd edn. Springer, New York (1998)
11. Fumagalli, I., Manzoni, A., Parolini, N., Verani, M.: Reduced basis approximation and a posteriori error estimates for parametrized elliptic eigenvalue problems. ESAIM: Math. Model. Numer. Anal. **50**, 1857–1885 (2016)
12. Gerner, A.L., Veroy, K.: Certified reduced basis methods for parametrized saddle point problems. SIAM J. Sci. Comput. **34**(5), A2812–A2836 (2012)
13. Glas, S., Urban, K.: On non-coercive variational inequalities. SIAM J. Numer. Anal. **52**, 2250–2271 (2014)
14. Glas, S., Urban, K.: Numerical investigations of an error bound for reduced basis approximations of noncoercice variational inequalities. IFAC-PapersOnLine **48**(1), 721–726 (2015)
15. Haasdonk, B., Salomon, J., Wohlmuth, B.: A reduced basis method for parametrized variational inequalities. SIAM J. Numer. Anal. **50**, 2656–2676 (2012)
16. Hesch, C., Betsch, P.: Isogeometric analysis and domain decomposition methods. Comput. Methods Appl. Mech. Eng. **213–216**, 104–112 (2012)
17. Horger, T., Wohlmuth, B., Dickopf, T.: Simultaneous reduced basis approximation of parameterized elliptic eigenvalue problems. ESAIM: Math. Model. Numer. Anal. **51**, 443–465 (2017)
18. Hughes, T.J.R., Cottrell, J.A., Bazilevs, Y.: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. Comput. Methods. Appl. Mech. Eng. **194**, 4135–4195 (2005)
19. Hughes, T.J.R., Evans, J.A., Reali, A.: Finite element and NURBS approximations of eigenvalue, boundary-value, and initial-value problems. Comput. Methods Appl. Mech. Eng. **272**, 290–320 (2014)
20. Iapichino, L., Quarteroni, A., Rozza, G., Volkwein, S.: Reduced basis method for the Stokes equations in decomposable domains using greedy optimization. In: European Conference Mathematics in Industry, ECMI 2014, pp. 1–7 (2014)
21. Jansson, E.V.: Violin frequency response – bridge mobility and bridge feet distance. Appl. Acoust. **65**(12), 1197–1205 (2004)
22. Lovgren, A., Maday, Y., Ronquist, E.: A reduced basis element method for the steady Stokes problem. Math. Model. Numer. Anal. **40**, 529–552 (2006)
23. Machiels, L., Maday, Y., Oliveira, I.B., Patera, A.T., Rovas, D.V.: Output bounds for reduced-basis approximations of symmetric positive definite eigenvalue problems. C.R. Acad. Sci., Paris, Sér. I **331**(2), 153–158 (2000)
24. Milani, R., Quarteroni, A., Rozza, G.: Reduced basis method for linear elasticity problems with many parameters. Comput. Methods Appl. Mech. Eng. **197**(51–52), 4812–4829 (2008)
25. Negri, F., Manzoni, A., Rozza, G.: Reduced basis approximation of parametrized optimal flow control problems for the Stokes equations. Comput. Math. Appl. **69**(4), 319–336 (2015)

26. Pau, G.: Reduced-basis method for band structure calculations. Phys. Rev. E **76**, 046704 (2007)
27. Pau, G.: Reduced basis method for simulation of nanodevices. Phys. Rev. B **78**, 155425 (2008)
28. Quarteroni, A.: Numerical Models for Differential Problems, MS&A, vol. 8, 2nd edn. Springer, Milan (2014)
29. Quarteroni, A., Manzoni, A., Negri, F.: Reduced Basis Methods for Partial Differential Equations. An Introduction. Springer, New York (2015)
30. Rand, O., Rovenski, V.: Analytical Methods in Anisotropic Elasticity: With Symbolic Computational Tools. Birkhäuser, Boston (2007)
31. Ranz, T.: Ein feuchte- und temperaturabhängiger anisotroper Werkstoff: Holz. Beiträge zur Materialtheorie. Universität der Bundeswehr München (2007)
32. Rozza, G., Veroy, K.: On the stability of the reduced basis method for Stokes equations in parametrized domains. Comput. Methods. Appl. Mech. Eng. **196**, 1244–1260 (2007)
33. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Application to transport and continuum mechanics. Arch. Comput. Methods Eng. **15**(3), 229–275 (2008)
34. Rozza, G., Huynh, D.B.P., Manzoni, A.: Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants. Numer. Math. **125**(1), 115–152 (2013)
35. Seitz, A., Farah, P., Kremheller, J., Wohlmuth, B., Wall, W., Popp, A.: Isogeometric dual mortar methods for computational contact mechanics. Comput. Methods Appl. Mech. Eng. **301**, 259–280 (2016)
36. Vallaghe, S., Huynh, D.P., Knezevic, D.J., Nguyen, T.L., Patera, A.T.: Component-based reduced basis for parametrized symmetric eigenproblems. Adv. Model. Simul. Eng. Sci. **2** (2015)
37. Woodhouse, J.: On the "bridge hill" of the violin. Acta Acust. United Acust. **91**(1), 155–165 (2005)

Chapter 7
Reduced Basis Approximations for Maxwell's Equations in Dispersive Media

Peter Benner and Martin Hess

Abstract Simulation of electromagnetic and optical wave propagation in, e.g. water, fog or dielectric waveguides requires modeling of linear, temporally dispersive media. Using a POD-greedy and ID-greedy sampling driven by an error indicator, we seek to generate a reduced model which accurately captures the dynamics over a wide range of parameters, modeling the dispersion. The reduced basis model reduction reduces the model order by a factor of more than 20, while maintaining an approximation error of significantly less than 1% over the whole parameter range.

7.1 Modeling Maxwell's Equations in Temporally Dispersive Media

The time-dependent Maxwell's equations in hyperbolic form, also termed the high-frequency approximation, is given in second order form in the electric field E as

$$\nabla \times \frac{1}{\mu} \nabla \times E + \varepsilon \frac{\partial^2 E}{\partial t^2} = -\frac{\partial j_i}{\partial t}, \tag{7.1}$$

with the material parameters permeability $\mu = \mu_r \mu_0$ and permittivity $\varepsilon = \varepsilon_r \varepsilon_0$, and an impressed source current density j_i. While the relative permeability μ_r and the relative permittivity ε_r depend on the material properties, μ_0 and ε_0 are constants corresponding to free space.

A temporally dispersive medium assumes a time-dependent relative permittivity $\varepsilon_r = \varepsilon_r(t)$, depending on the history of the electric field strength [9, 11]. Such a

P. Benner
Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany
e-mail: benner@mpi-magdeburg.mpg.de

M. Hess (✉)
Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg, Sandtorstr. 1, 39106 Magdeburg, Germany
e-mail: mhess@sissa.it

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_7

relation needs to be taken into account when the propagation of an electromagnetic pulse through certain media, such as water or fog, is simulated. A particular application in medical imaging is the noninvasive interrogation of the interior of tissues by electromagnetic waves. The approach followed here assumes a medium of polar molecules with a permanent dipole moment. A concrete example is the water molecule H_2O, where the hydrogen atoms are attached at a certain angle to the oxygen atom. While the molecule as a whole is neutrally charged, there is a permanent dipole moment due to the particular angle.

Thus, the Maxwell's equations (7.1) are altered by taking a relaxation polarization into account. Considering liquid or solid dielectrics with polar molecules, the molecules reply to the applied external field by rotating, i.e., rotating such that the dipole moment is in sync with the external field. This causes friction, which in turn leads to an exponential damping of the electromagnetic pulse [1, 3, 11]. Since the wave patterns are similar for different parameter configurations, we expect this problem to be well-suited for projection-based model reduction. The dispersive property is incorporated by replacing $\varepsilon_0\varepsilon_r E(t,x,y)$ with

$$\varepsilon_0\varepsilon_\infty E(t,x,y) + \varepsilon_0 \int_{-\infty}^{t} E(t-\tau,x,y)\chi(\tau)d\tau \tag{7.2}$$

with susceptibility χ and spatial coordinates x and y. Here, a single pole expansion of the susceptibility in frequency domain is assumed, called a Debye relaxation, with relative permittivity at low-frequency limit ε_s, relative permittivity at high-frequency limit ε_∞ and relaxation time τ:

$$\chi(\omega,x,y) = \frac{(\varepsilon_s - \varepsilon_\infty)}{\imath\omega\tau + 1}. \tag{7.3}$$

Making use of the exponential decay of χ in time, the convolution integral can be computed efficiently, see [9]. Here, the approach is to derive an auxiliary differential equation for the polarization [6, 9], which avoids computation of the convolution integral. Define $P(t,x,y)$ as the relaxation polarization (in the following referred to as just the 'polarization')

$$P(t,x,y) = \varepsilon_0 \int_{-\infty}^{t} E(t-\tau,x,y)\chi(\tau,x,y)d\tau, \tag{7.4}$$

which leads to

$$\tau\partial_t P + P = \varepsilon_0(\varepsilon_s - \varepsilon_\infty)E, \tag{7.5}$$

using the single pole expansion of χ, such that the coupled system of E and P is given as

$$\nabla \times \left(\frac{1}{\mu_0} \nabla \times E \right) + \varepsilon_0 \varepsilon_\infty \partial_t^2 E = f - \partial_t^2 P, \tag{7.6}$$

$$\partial_t P + \frac{1}{\tau} P = \frac{\varepsilon_0 (\varepsilon_s - \varepsilon_\infty)}{\tau} E, \tag{7.7}$$

with polarization P and an input source denoted by f. This is called the Debye model of orientational polarization, or Maxwell-Debye model for short.

Typical material parameter values for water are $\varepsilon_\infty = 1.80, \varepsilon_s = 81.00, \tau = 9.400 \times 10^{-12}$ s and for foam $\varepsilon_\infty = 1.01, \varepsilon_s = 1.16, \tau = 6.497 \times 10^{-10}$ s.

Section 7.2 details how the coupled system of Eqs. (7.6)–(7.7) is solved numerically, while Sect. 7.3 explains the model reduction procedures. Section 7.4 provides numerical results and Sect. 7.5 concludes our findings.

7.2 Simulation of Maxwell's Equations in Temporally Dispersive Media

The discretization of (7.6)–(7.7) is done with Nédélec finite elements of first order [14]. As a test case, the 2-dimensional unit square in the x-y-plane is chosen, corresponding to a physical domain of 1 m-by-1 m. The finite element method is implemented in MATLAB from first principles. The uniform triangulation of the domain results in 9680 edge-based degrees of freedom, which serve as projection space for the full-order model in the discretized electric field $E^{\mathcal{N}}$ as well as the discretized polarization $P^{\mathcal{N}}$. The boundary is a zero Dirichlet boundary (also called PEC—perfectly electric conducting), enforced by setting the appropriate degrees of freedom to zero. To achieve a broadband input source, a Gaussian pulse is used to excite the system. In particular, the curl of a Gaussian in the z-direction is used, which physically corresponds to a magnetic field present perpendicular to the computational domain. Define the Gaussian pulse $G : \mathbb{R}^2 \to \mathbb{R}$ as

$$G(x, y) = \frac{1}{\sqrt{|\Sigma|(2\pi)^2}} \exp(-\frac{1}{2}((x, y)^T - \mu)^T \Sigma^{-1} ((x, y)^T - \mu)), \tag{7.8}$$

with mean in the center of the domain $\mu = (0.5, 0.5)^T$ and covariance matrix $\Sigma = \begin{pmatrix} 0.1 & 0 \\ 0 & 0.1 \end{pmatrix}$, then the (vector-valued) spatial source term $f_1(x, y)$ is

$$f_1(x, y) = \nabla \times \begin{pmatrix} 0 \\ 0 \\ G(x, y) \end{pmatrix} = \begin{pmatrix} \partial_y G(x, y) \\ -\partial_x G(x, y) \\ 0 \end{pmatrix}. \tag{7.9}$$

The spatial source term is multiplied with the second derivative of a Gaussian in time $f_2(t)$, such that the input source is $f = f(x,y,t) = f_1(x,y)f_2(t)$. The total time interval is $[0,5]$ ns, divided into 500 timesteps of 10 ps each.

The time-stepping is realized by a Runge-Kutta-Nyström scheme in the electric field and an explicit Euler in the polarization and applied to the semi-discretized system with mass matrix M and the curl-curl matrix S. The Runge-Kutta-Nyström scheme [4] makes use of the particular form $\ddot{E}^{\mathscr{N}} = \mathscr{F}(t, E^{\mathscr{N}})$ found in (7.6) by a simple transformation. The Eqs. (7.6) and (7.7) are solved in turn, i.e. assuming an initial condition of $E^{\mathscr{N}}(t,x,y) = 0$ and $P^{\mathscr{N}}(t,x,y) = 0$, first (7.6) is solved and then the time derivative of (7.7) is solved for $\partial_t^2 P^{\mathscr{N}}$ with $\partial_t E^{\mathscr{N}}$ plugged in. The solution for $\partial_t^2 P^{\mathscr{N}}$ is then used in (7.6) for the next timestep. The Runge-Kutta-Nyström scheme used here computes for each timestep t_k

$$k_1 = \mathscr{F}(t_k, E^{\mathscr{N}}(t_k) + \frac{3+\sqrt{3}}{6}\Delta_t \partial_t E^{\mathscr{N}}(t_k)), \tag{7.10}$$

$$k_2 = \mathscr{F}(t_k, E^{\mathscr{N}}(t_k) + \frac{3-\sqrt{3}}{6}\Delta_t \partial_t E^{\mathscr{N}}(t_k) + \frac{2-\sqrt{3}}{12}\Delta_t^2 k_1), \tag{7.11}$$

$$k_3 = \mathscr{F}(t_k, E^{\mathscr{N}}(t_k) + \frac{3+\sqrt{3}}{6}\Delta_t \partial_t E^{\mathscr{N}}(t_k) + \frac{\sqrt{3}}{6}\Delta_t^2 k_2), \tag{7.12}$$

$$\begin{aligned} E^{\mathscr{N}}(t_{k+1}) &= E^{\mathscr{N}}(t_k) \\ &+ \Delta_t \partial_t E^{\mathscr{N}}(t_k) + \Delta_t^2 (\frac{5-3\sqrt{3}}{24}k_1 + \frac{3+\sqrt{3}}{12}k_2 + \frac{1+\sqrt{3}}{24}k_3), \end{aligned} \tag{7.13}$$

$$\partial_t E^{\mathscr{N}}(t_{k+1}) = \partial_t E^{\mathscr{N}}(t_k) + \Delta_t (\frac{3-2\sqrt{3}}{12}k_1 + \frac{1}{2}k_2 + \frac{3+2\sqrt{3}}{12}k_3), \tag{7.14}$$

such that $\partial_t^2 E^{\mathscr{N}}(t_{k+1})$ can be computed by evaluating $\mathscr{F}$ at t_{k+1}, but this is actually not necessary, since $\partial_t^2 E^{\mathscr{N}}(t_{k+1})$ is not required in (7.7). Since the relaxation time is in the range of nanoseconds, the time-stepping is chosen as $\Delta_t = 1\times10^{-11}$ s $= 10$ ps with a total number of timesteps $n_T = 500$.

Example trajectories are shown in Figs. 7.1 and 7.2. Due to the smaller permittivities in Fig. 7.1, the propagation velocity is larger and the fields are not as strongly damped as in Fig. 7.2. In general, different parameter choices for ε_s, ε_∞ and τ mainly change the propagation velocity and amplitudes.

7.3 Reduced Basis Parametric Model Order Reduction

Parametric model order reduction (PMOR) aims at reducing the computational effort in many-query and real-time tasks. A recent survey of PMOR techniques can be found in [2]. The PMOR technique applied in this work is the reduced basis method (RBM) [12]. Several studies of model order reduction techniques for

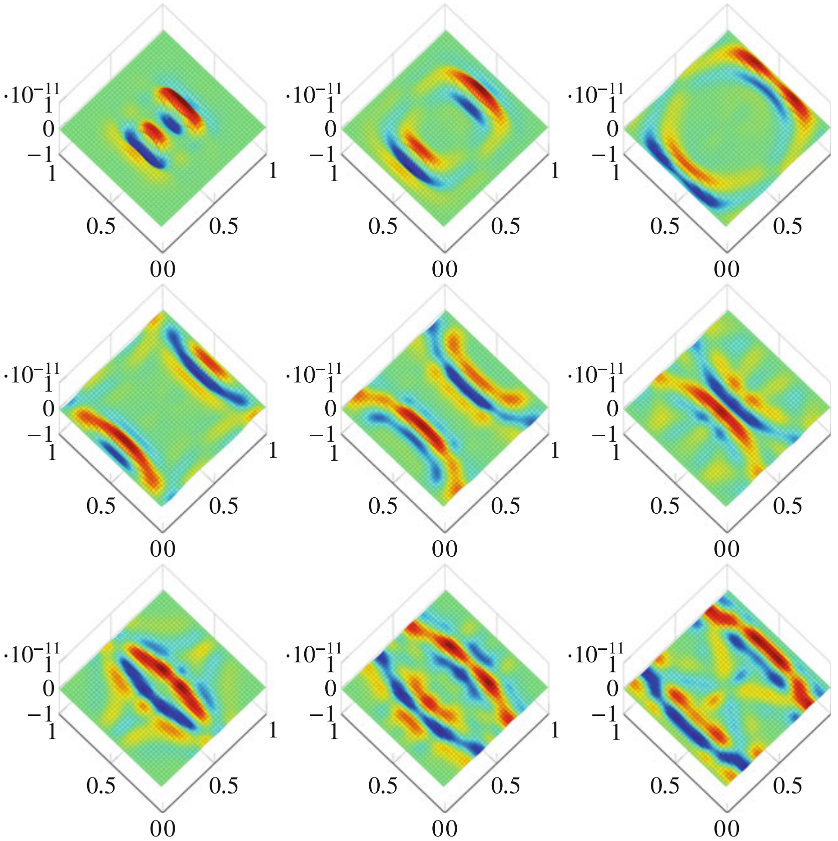


Fig. 7.1 Snapshots of one component of the trajectory using $\varepsilon_\infty = 1$, $\varepsilon_s - \varepsilon_\infty = 0.5$ and $\tau = 1{\times}10^{-9}$. *From top left to bottom right*, timesteps 100, 150, 200, 250, 300, 350, 400, 450 and 500

Maxwell's equations have been performed successfully [8, 10, 13], but the dispersive case is not covered so far. A low-order model is determined by the RBM from a few large-scale solves at judiciously chosen parameter locations. A particular role play error indicators, which determine the parameter locations of choice. At these parameter locations, a full order trajectory is computed and condensed to the most dominant modes by a matrix decomposition. The most significant modes are then stored in a projection matrix X_N. Since it is a coupled problem in the electric field $E^{\mathscr{N}}$ and the polarization $P^{\mathscr{N}}$, a compound reduced basis X_N is formed by independently condensing both trajectories. To ensure numerical stability, the compound reduced basis space needs to be orthonormalized separately. Since the electric field and the polarization are differently scaled by a factor of about $1{\times}10^{-6}$, the columns are normalized before orthonormalization. A Ritz-Galerkin projection is used, such that the large scale matrices A^i are projected as $A_N^i = X_N^T A^i X_N$.

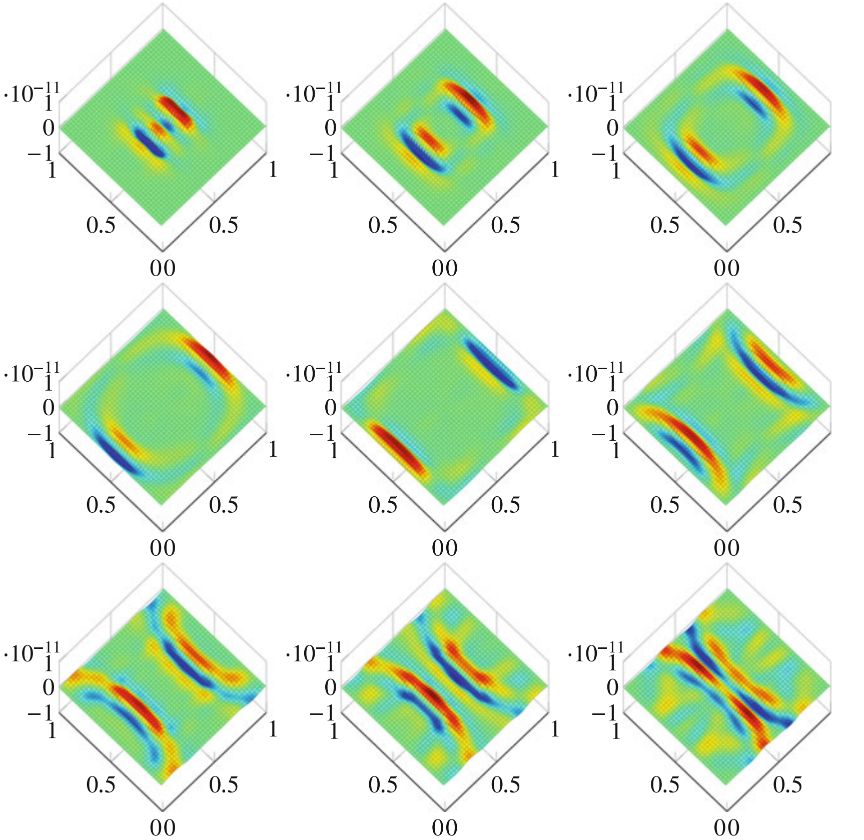


Fig. 7.2 Snapshots of one component of the trajectory using $\varepsilon_\infty = 2$, $\varepsilon_s - \varepsilon_\infty = 1.5$ and $\tau = 1\times 10^{-8}$. *From top left to bottom right*, timesteps 100, 150, 200, 250, 300, 350, 400, 450 and 500

The same procedure is applied to the mass matrix M and the curl-curl matrix S. The model is parametrized in the relaxation time $\tau \in [1\times 10^{-10}, 1\times 10^{-6}]$ with 12 logarithmically spaced samples and $\Delta_\varepsilon = \varepsilon_s - \varepsilon_\infty \in [0.5, 10]$ with 10 uniformly spaced samples, thus defining the 2-dimensional sampled parameter domain Ξ_{train}. A point in Ξ_{train} is denoted by the parameter vector ν. Assuming an initial reduced order model has been computed with reduced order solution $E_N(\cdot, \nu)$, it holds

$$\Delta_N(\nu) = \|E^{\mathcal{N}}(\cdot, \nu) - E_N(\cdot, \nu)\| \leq \sum_{k=1}^{n} a^k(\nu) b_{n-k}(\nu), \tag{7.15}$$

where

$$a(\nu) = \sup_{n\in\{1,\ldots,n_T\}} \sup_{w\neq y} \frac{\|w-y\|}{\|f(w,t_n;\nu) - f(y,t_n;\nu)\|}, \tag{7.16}$$

is the inverse Lipschitz constant and $b_n(\nu)$ is the residual at timestep n.

The typical procedure is a greedy-max strategy, where the maximum of $\Delta_N(\nu)$ over the sampled parameter domain is chosen for another large-scale solve to enrich the reduced basis. However, rigorous error estimation requires an upper bound on the inverse Lipschitz constant, which is not feasible by current methods. Thus, an error indicator is used by setting $a = 1$. Another option is to use finite volume discretizations as in [7], where the Lipschitz constant does not appear in the error estimate.

Time-dependent problems typically employ the POD-greedy technique, which uses a proper orthogonal decomposition (POD) to condense the time trajectory. For comparison, we also show results using an ID-greedy procedure, where an interpolatory decomposition (ID) is used.

7.3.1 POD-Greedy Algorithm

The POD-Greedy approach is a well-established technique for model reduction of time-dependent problems [7]. The POD performs a singular value decomposition (SVD) on the orthogonal complement of the newly computed trajectory with respect to the current projection basis X_N. The projection onto the space X_N is denoted by the operator Π_{X_N} in the POD-Greedy algorithm. The modes corresponding to the largest singular values are then appended to the projection basis. This ensures that the most important information on the trajectory is appended due to the best approximation property of the SVD.

The POD compression in time offers some tuning options. Either a fixed number of modes can be appended to the projection basis, or a number of modes corresponding to a percentage of the sum of the singular values. A high percentage, such as 99%, is typically sufficient to resolve the trajectory accurately. This will be used here, such that the modes corresponding to the largest singular values are chosen until the sum of the associated singular values reaches 99% of the sum of all singular values.

When the time trajectory is large, the POD can become infeasible to compute. A compression of the trajectory is thus useful and can be achieved by an adaptive snapshot selection [15]. Successively removing vectors from the trajectory, when the angle to the last chosen vector is below a threshold angle, can significantly reduce the size of the trajectory, without impacting the approximation accuracy. A variation of this is looking at the angle between the current vector and the whole subspace, which has already been chosen.

Algorithm 1 POD-Greedy algorithm
INPUT: sampled parameter domain Ξ_{train}, maximum iteration number k_{MAX}
OUTPUT: POD-Greedy samples S_N, projection space $X_{N(k_{MAX})}$

```
Choose $\nu_1 \in \Xi_{train}$ arbitrarily
Solve for $E^{\mathcal{N}}(t_i;\nu_1)$ and $P^{\mathcal{N}}(t_i;\nu_1)$, where $i = 1,\dots,n_T$
Set $S_1 = \{\nu_1\}$
Set $N(0) = 0$
Set $k = 1$
POD of trajectories for $E^{\mathcal{N}}(t_i;\nu_1)$ and $P^{\mathcal{N}}(t_i;\nu_1)$ gives initial compound projection basis $X_{N(k)}$
Let $\ell(k)$ be the number of added basis vectors (w.r.t. the prescribed tolerance), then $N(k) = N(k-1) + \ell(k)$
while $k < k_{MAX}$ do
    Set $k = k + 1$;
    Set $\nu_k = \arg\max_{\nu \in \Xi_{train}} \Delta_{N(k)}(\nu)$
    Set $S_k = S_{k-1} \cup \{\nu_k\}$
    Solve model for $E^{\mathcal{N}}(t_i;\nu_k)$ and $P^{\mathcal{N}}(t_i;\nu_k)$
    $e_E(t_i) = E^{\mathcal{N}}(t_i;\nu_k) - \Pi_{X_N} E^{\mathcal{N}}(t_i;\nu_k)$, where $i = 1,\dots,n_T$
    $e_P(t_i) = P^{\mathcal{N}}(t_i;\nu_k) - \Pi_{X_N} P^{\mathcal{N}}(t_i;\nu_k)$, where $i = 1,\dots,n_T$
    POD of trajectories $e_E(t_i)$ and $e_P(t_i)$ and append modes to $X_{N(k)}$
    Let $\ell(k)$ be the number of added basis vectors (w.r.t. the prescribed tolerance), then $N(k) = N(k-1) + \ell(k)$
end while
```

7.3.2 ID-Greedy Algorithm

Since the singular value decomposition (SVD) in the POD step might be costly, an interpolatory decomposition (ID) is considered as an alternative, [5]. The potential advantage is that computation times are lower than for an SVD. On the other hand, the interpolatory decomposition does not generate orthonormal matrices.

The interpolatory decomposition of a matrix $A \in R^{m\times n}$ is a randomized decomposition into $U \in R^{m\times k}$, $B \in R^{k\times k}$ and $V \in R^{m\times k}$ as

$$A \approx U \circ B \circ V^T, \tag{7.17}$$

where B is a $k \times k$ submatrix of A. The norm of U and V is close to one and both matrices contain a $k \times k$ submatrix, see [5] for more details. It expresses each of the columns of A as a linear combination of k selected columns of A and analogously for the rows. This selection defines the $k \times k$ submatrix B of A, and in the resulting system of coordinates, the action of A is represented by the action of its submatrix B. Either the order k or an error tolerance is specified for its operation. Here, the ID-greedy appends at most 20 basis vectors in each iteration.

The ID-greedy algorithm essentially only differs from the POD-greedy in that the interpolative decomposition replaces the POD.

Algorithm 2 ID-Greedy algorithm
INPUT: sampled parameter domain Ξ_{train}, maximum iteration number k_{MAX}
OUTPUT: ID-Greedy samples S_N, projection space $X_{N(k_{MAX})}$

```
Choose $\nu_1 \in \Xi_{train}$ arbitrarily
Solve for $E^{\mathcal{N}}(t_i;\nu_1)$ and $P^{\mathcal{N}}(t_i;\nu_1)$, where $i = 1,\dots,n_T$
Set $S_1 = \{\nu_1\}$
Set $N(0) = 0$
Set $k = 1$
ID of trajectories for $E^{\mathcal{N}}(t_i;\nu_1)$ and $P^{\mathcal{N}}(t_i;\nu_1)$ gives initial compound projection basis $X_{N(1)}$
Let $\ell(k)$ be the number of added basis vectors (w.r.t. the prescribed tolerance), then $N(k) = N(k-1) + \ell(k)$
while $k < k_{MAX}$ do
    Set $k = k + 1$;
    Set $\nu_k = \arg\max_{\nu \in \Xi_{train}} \Delta_{N(k)}(\nu)$
    Set $S_k = S_{k-1} \cup \{\nu_k\}$
    Solve model for $E^{\mathcal{N}}(t_i;\nu_k)$ and $P^{\mathcal{N}}(t_i;\nu_k)$
    $e_E(t_i) = E^{\mathcal{N}}(t_i;\nu_k) - \Pi_{X_N} E^{\mathcal{N}}(t_i;\nu_k)$, where $i = 1,\dots,n_T$
    $e_P(t_i) = P^{\mathcal{N}}(t_i;\nu_k) - \Pi_{X_N} P^{\mathcal{N}}(t_i;\nu_k)$, where $i = 1,\dots,n_T$
    ID of trajectories $e_E(t_i)$ and $e_P(t_i)$ and append modes to $X_{N(k)}$
    Let $\ell(k)$ be the number of added basis vectors (w.r.t. the prescribed tolerance), then $N(k) = N(k-1) + \ell(k)$
end while
```

7.4 Numerical Results

The trajectory of a single degree of freedom for the parameters $\Delta_\varepsilon = 0.1$ and $\tau = 1.23\times10^{-9}$ is chosen for reference, which is a parameter configuration, that was not chosen explicitly in either the POD-greedy or ID-greedy procedure. In Figs. 7.3 and 7.4, the error in that degree of freedom is shown with respect to increasing model sizes. It indicates, that the POD-greedy and the ID-greedy deliver accurate reduced order models of comparable size. Note that due to the employed version of the greedy procedures, the number of added basis vectors per iteration varies and can get as large as 29. Thus the explicit dependence of the reduced dimension N on the iteration number k in the algorithms.

Over the whole parameter domain, there is a smooth decay in the maximum error as shown in Fig. 7.5, indicating exponential convergence speed. The computation times[1] of using a POD-greedy reduced order model are shown in Table 7.1. Using a model of dimension 123, the compute time of a trajectory reduces by a factor of 35.

[1] All computations were done on a Intel(R) Core(TM)2 Quad CPU Q6700 @ 2.66GHz desktop machine with 8GB RAM, running Ubuntu 12.04.5 LTS and MATLAB R2012b.

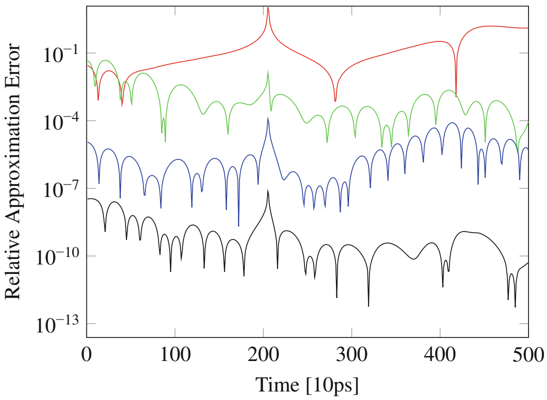


Fig. 7.3 Error in the reference trajectory for POD-greedy generated models of size 9 (*red*), 21 (*green*), 79 (*blue*) and 123 (*black*)

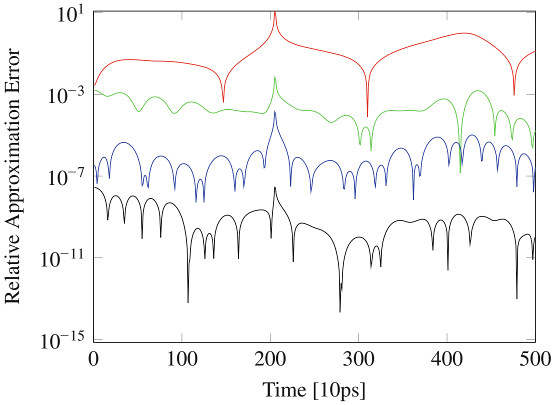


Fig. 7.4 Error in the reference trajectory for ID-greedy generated models of size 20 (*red*), 40 (*green*), 79 (*blue*) and 125 (*black*)

To better access the error decay of both methods, the algorithms are run again with different choices of parameters, such that more intermediate models are computed. The results are shown in Fig. 7.6. In particular, the interpolatory decomposition appends at most six basis vectors in each greedy iteration and the POD appends singular vectors corresponding to 80% of the sum of the singular vectors. With this choice of tuning options for the POD-greedy and ID-greedy, the greedy chooses the same parameter vector multiple times. This results in more intermediate models but also creates computational overhead.

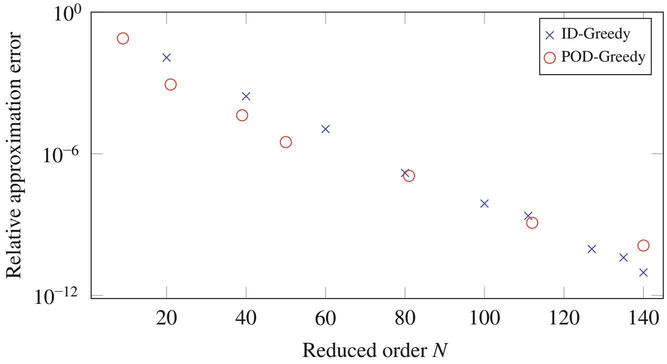


Fig. 7.5 Comparison of SVD (*red*) and ID (*blue*) generated models. Shown is the maximum relative error (mean in time) in the electric field over the sampled parameter domain

Table 7.1 Comparison of computation times of full order and reduced order models

Model size	9680	123	79	21	9
Time	43.5 s	1.24 s	0.67 s	0.12 s	0.07 s

The full order model is of size 9680

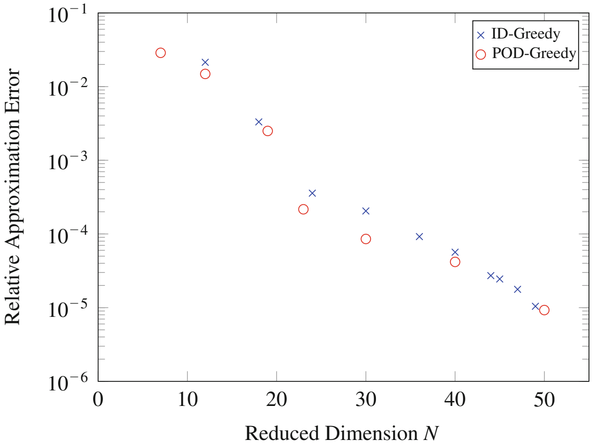


Fig. 7.6 Comparison of SVD (*red*) and ID (*blue*) generated models with more intermediate models in contrast to Fig. 7.5. Shown is the maximum relative error (mean in time) in the electric field over the sampled parameter domain

In this model setup, the dominant offline time is in the computation of residuals after each basis enrichment step, see Table 7.2. However, with an increasing number of timesteps, the matrix decompositions will become more expensive. Table 7.3 shows the compute times of the matrix decompositions. It shows that with increasing number of timesteps, adaptive snapshot selection [15] might become necessary.

Table 7.2 Comparison of POD-greedy and ID-greedy computation times for different maximum approximation errors, the number of greedy iterations is shown in brackets

Relative error	POD-greedy time (greedy iterations)	ID-greedy time (greedy iterations)
1×10^{-3}	203 s (1)	266 s (1)
1×10^{-5}	982 s (3)	717 s (2)
1×10^{-7}	1622 s (4)	2313 s (4)

Table 7.3 Comparison of SVD (POD) and ID computation times for different number of timesteps

Method	500 timesteps	1500 timesteps	3000 timesteps
SVD	2.9 s	10.9 s	26.6 s
ID	0.9 s	4.4 s	13.2 s

7.5 Conclusion

This is the first application of model reduction to Maxwell's equations in dispersive media, to the best of our knowledge. Since the parametric variations mainly influence propagation velocity and amplitudes, this problem is well suited for parametric model reduction. In short, if a single trajectory is well resolved, then this also extends to other parameter locations. A reduced model order of 50 shows less then 0.1% approximation error from the full order model of size 9680.

A reason for that is that the Debye relaxation does introduce an exponential damping and does not show trailing waves. A different relaxation under the Lorentz-Lorenz relation or Drude model [11] might prove more difficult for the model reduction.

Acknowledgements The authors thank Prof. Jan Hesthaven (EPFL) for helpful advice and fruitful discussion.

This work is supported by the collaborative project nanoCOPS, Nanoelectronic COupled Problems Solutions, supported by the European Union in the FP7-ICT-2013-11 Program under Grant Agreement Number 619166.

References

1. Banks, H.T., Bokil, V.A., Gibson, N.L.: Analysis of stability and dispersion in a finite element method for Debye and Lorentz dispersive media. Numer. Methods Partial Differ. Equ. **25**(4), 885–917 (2009)
2. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. **57**(4), 483–531 (2015)
3. Bidégaray-Fesquet, B.: Stability of FDTD schemes for Maxwell-Debye and Maxwell-Lorentz equations. SIAM J. Numer. Anal. **46**(5), 2551–2566 (2008)
4. Calvo, M.P., Sanz-Serna, J.M.: Order conditions for canonical Runge-Kutta-Nyström methods. BIT Numer. Math. **32**(1), 131–142 (1992)

5. Cheng, H., Gimbutas, Z., Martinsson, P.G., Rokhlin, V.: On the compression of low rank matrices. SIAM J. Sci. Comput. **26**(4), 1389–1404 (2005)
6. Goswami, C., Mukherjee, S., Karmakar, S., Pal, M., Ghatek, R.: FDTD modeling of Lorentzian DNG metamaterials by auxiliary differential equation method. J. Electromagn. Anal. Appl. **6**(5), 106–114 (2014)
7. Haasdonk, B., Ohlberger, M.: Reduced basis method for finite volume approximations of parametrized linear evolution equations. ESAIM: Math. Model. Numer. Anal. **42**(2), 277–302 (2008)
8. Henneron, T., Clenet, S.: Model order reduction of non-linear magnetostatic problems based on POD and DEI methods. IEEE Trans. Magn. **50**(2), 33–36 (2014). doi:10.1109/TMAG.2013.2283141
9. Jin, J.M.: Theory and Computation of Electromagnetic Fields. Wiley, New York (2011)
10. Jung, N., Patera, A., Haasdonk, B., Lohmann, B.: Model order reduction and error estimation with an application to the parameter-dependent eddy current equation. Math. Comput. Modell. Dyn. Syst. **17**, 561–582 (2011)
11. Oughstun, K.E.: Electromagnetic and Optical Pulse Propagation 1: Spectral Representations in Temporally Dispersive Media. Electromagnetic and Optical Pulse Propagation. Springer, New York (2006)
12. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Arch. Comput. Meth. Eng. **15**, 229–275 (2008)
13. Schmidthaeusler, D., Clemens, M.: Low-order electroquasistatic field simulations based on proper orthogonal decomposition. IEEE Trans. Magn. **48**, 567–570 (2012)
14. Schneebeli, A.: An H(curl;Ω)-conforming FEM: Nédélec's elements of first type. Technical Report (2003)
15. Zhang, Y., Feng, L., Li, S., Benner, P.: Accelerating PDE constrained optimization by the reduced basis method: application to batch chromatography. Int. J. Numer. Methods Eng. **104**(11), 983–1007 (2015)

Chapter 8
Offline Error Bounds for the Reduced Basis Method

Robert O'Connor and Martin Grepl

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

8.1 Introduction

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

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

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_8

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

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

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

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

It is this lack of guaranteed offline certification in combination with the real-time context which motivated the development in this paper. Our goal is to develop an approach which allows us to rigorously guarantee a certain accuracy of the reduced basis approximation over the entire parameter domain $\mathscr{D}$, and not just over the train set Ξ. Our method can be considered an "add-on" to the standard greedy algorithm: in addition to the reduced basis approximation and associated a posteriori error bounds we also evaluate the sensitivity information and their error bounds on a finite train set. Given these quantities, we can then bound the error at any parameter value in the domain and thus bound the accuracy of the reduced basis approximation over $\mathscr{D}$—we use the term "offline bound" for this approach. In that way reduced basis models can be guaranteed to satisfy error tolerances for real-time applications. Obviously, our approach incurs an additional offline cost (see Sect. 8.4) and is thus not useful for applications where one can go back to the offline stage at will and refine the reduced basis approximation at any time. However, if an offline-guaranteed accuracy is essential for the application, the added offline cost may be the only choice and thus acceptable. We note that our results may also be interesting in many-query contexts because they allow us to perform error bounding in the offline stage, reducing the workload in the online stage.

8.2 Problem Statement

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

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

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

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

and coercive,

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

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

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

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

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

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

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

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

8.2.1 Sensitivity Derivatives

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

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

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

8.3 The Reduced Basis Method

8.3.1 Approximation

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

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

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

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

8.3.2 A Posteriori Error Estimation

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

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

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

$$\begin{aligned} r_\eta(v; \mu) \equiv \sum_{q=1}^{Q_f} \nabla_\eta \Theta_f^q(\mu) f^q(v) - a\left(\nabla_\eta u_N(\mu), v; \mu\right) \\ - \sum_{q=1}^{Q_a} \nabla_\eta \Theta_a^q(\mu) a^q(u_N(\mu), v), \qquad \forall v \in X. \end{aligned} \tag{8.9}$$

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

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

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

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

and

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

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

8.4 Offline Error Bounds

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

8.4.1 Bounding the Difference Between Solutions

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

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

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

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

$$a(d(\mu,\omega), v; \omega) = \sum_{q=1}^{Q_f} \Big(\Theta_f^q(\mu) - \Theta_f^q(\omega) \Big) f^q(v) \\ - \sum_{q=1}^{Q_a} \Big(\Theta_a^q(\mu) - \Theta_a^q(\omega) \Big) a^q(u(\mu), v). \tag{8.13}$$

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

$$\alpha_{\mathrm{LB}}(\omega) \| d(\mu,\omega) \|_X^2 \le \| d(\mu,\omega) \|_X \left(\sum_{q=1}^{Q_f} \gamma_{f,q} \left| \Theta_f^q(\mu) - \Theta_f^q(\omega) \right| \right. \\ \left. + \| u(\mu) \|_X \sum_{q=1}^{Q_a} \gamma_{a,q} \left| \Theta_a^q(\mu) - \Theta_a^q(\omega) \right| \right). \tag{8.14}$$

Cancelling and rearranging terms gives the desired result. □

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

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

$$\| \nabla_\eta d(\mu,\omega) \|_X \le \sum_{q=1}^{Q_f} \frac{\gamma_{f,q}}{\alpha_{\mathrm{LB}}(\omega)} | \nabla_\eta \Theta_f^q(\mu) - \nabla_\eta \Theta_f^q(\omega) | \\ + \sum_{q=1}^{Q_a} \frac{\gamma_{a,q}}{\alpha_{\mathrm{LB}}(\omega)} \Big(| \Theta_a^q(\mu) - \Theta_a^q(\omega) | \| \nabla_\eta u(\mu) \|_X + | \nabla_\eta \Theta_a^q(\omega) | \| d(\mu,\omega) \|_X \\ + | \nabla_\eta \Theta_a^q(\mu) - \nabla_\eta \Theta_a^q(\omega) | \| u(\mu) \|_X \Big). \tag{8.15}$$

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

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

8.4.2 An Initial Offline Bound

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

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

$$\|e(\omega)\|_X \leq \Delta^0(\mu,\omega) \equiv \Delta(\mu) + \frac{2}{\alpha_{\mathrm{LB}}(\omega)}\left(\sum_{q=1}^{Q_f} \gamma_{f,q}\left|\Theta_f^q(\mu) - \Theta_f^q(\omega)\right|\right) + \frac{2\|u_N(\mu)\|_X + \Delta(\mu)}{\alpha_{\mathrm{LB}}(\omega)}\left(\sum_{q=1}^{Q_a} \gamma_{a,q}\left|\Theta_a^q(\mu) - \Theta_a^q(\omega)\right|\right). \tag{8.16}$$

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

$$e(\omega) = e(\mu) - d(\mu,\omega) + d_N(\mu,\omega). \tag{8.17}$$

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

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

8.4.3 Bounds Based on First-Order Sensitivity Derivatives

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

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

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

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

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

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

$$\begin{aligned}\|e(\omega)\|_X \leq \Delta^1(\mu,\omega) \equiv \Delta(\mu) + \Delta_\rho(\mu) + \frac{2}{\alpha_{LB}}\sum_{q=1}^{Q_f}\gamma_{f,q} I^q_{f,\nabla} \\ + \sum_{q=1}^{Q_a}\frac{\gamma_{a,q}}{\alpha_{LB}}\left(\Big(2\|u_N(\mu)\|_X + \Delta(\mu)\Big)\left(I^q_{a,\nabla} + \sum_{\bar{q}=1}^{Q_a}\frac{\gamma_{a,\bar{q}} I^{q,\bar{q}}_{a,a}}{\alpha_{LB}}\right)\right. \\ \left. + \Big(2\|\nabla_\rho u_N(\mu)\|_X + \Delta_\rho(\mu)\Big) I^q_a + \frac{2}{\alpha_{LB}}\sum_{\bar{q}=1}^{Q_f}\gamma_{f,\bar{q}} I^{q,\bar{q}}_{a,f}\right),\end{aligned} \tag{8.19}$$

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

$$I^q_{f,\nabla} \equiv \int_0^1 \Big|\nabla_\rho \Theta^q_f(\mu) - \nabla_\rho \Theta^q_f(\mu+\tau\rho)\Big|\, d\tau, \tag{8.20a}$$

$$I^q_{a,\nabla} \equiv \int_0^1 \big|\nabla_\rho \Theta^q_a(\mu) - \nabla_\rho \Theta^q_a(\mu+\tau\rho)\big|\, d\tau, \tag{8.20b}$$

$$I^q_a \equiv \int_0^1 \big|\Theta^q_a(\mu) - \Theta^q_a(\mu+\tau\rho)\big|\, d\tau, \tag{8.20c}$$

$$I^{q,\bar{q}}_{a,f} \equiv \int_0^1 \big|\nabla_\rho \Theta^q_a(\mu+\tau\rho)\big| \Big|\Theta^{\bar{q}}_f(\mu) - \Theta^{\bar{q}}_f(\mu+\tau\rho)\Big|\, d\tau, \tag{8.20d}$$

$$I^{q,\bar{q}}_{a,a} \equiv \int_0^1 \big|\nabla_\rho \Theta^q_a(\mu+\tau\rho)\big| \big|\Theta^{\bar{q}}_a(\mu) - \Theta^{\bar{q}}_a(\mu+\tau\rho)\big|\, d\tau. \tag{8.20e}$$

Proof We begin with the fundamental theorem of calculus

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

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

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

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

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

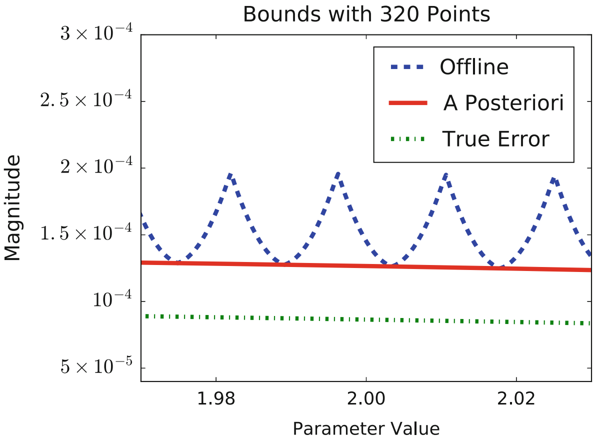


Fig. 8.1 Results using the first-order offline bounds

peaks of the blue dashed line in Fig. 8.1. In the numerical results presented below we therefore only plot the upper bound obtained by connecting these peaks.

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

8.5 Computing Coercivity Constants

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

Let us introduce the set

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

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

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

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

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

8.6 Numerical Results

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

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

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

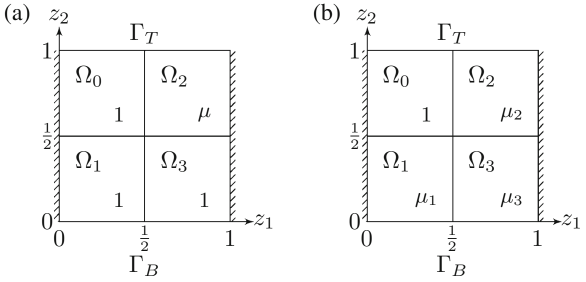


Fig. 8.2 2×2 thermal block model problem. (**a**) Thermal block with 1 parameter. (**b**) Thermal block with 3 parameter

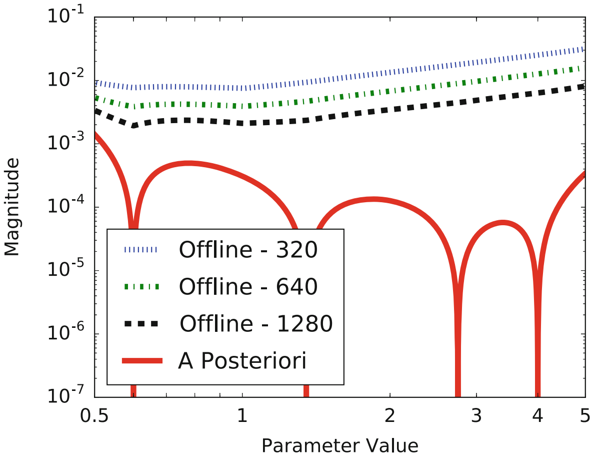


Fig. 8.3 Results using the zero-order offline bounds

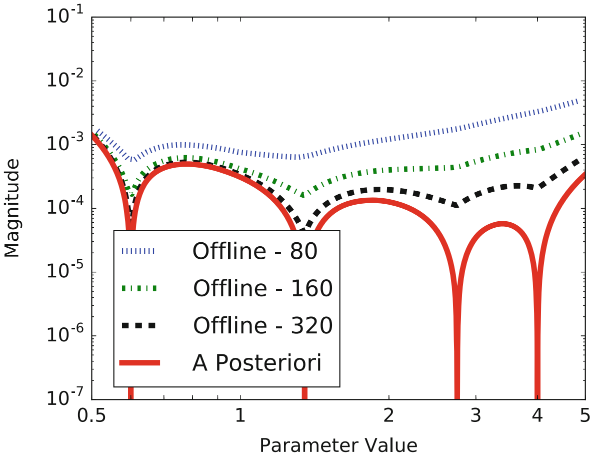


Fig. 8.4 Results using the first-order offline bounds

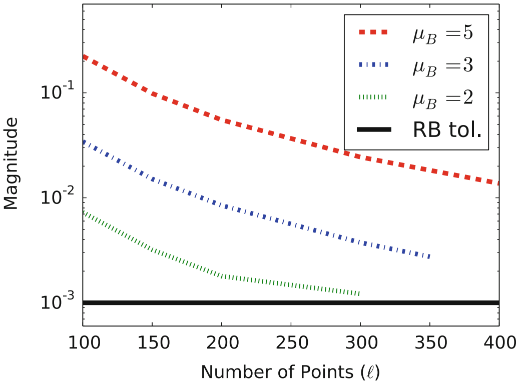
peaks. For all practical applications the quick variations are unimportant and only the peaks are of interest, since they represent the worst case upper bound.

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

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

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

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



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

8.7 Conclusions and Extensions

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

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

We have derived zero and first-order offline bounds. We expect that using higher-order bounds would produce better results and reduce the computational cost. It may also be interesting to tailor the reduced basis space to produce accurate approximations of not only the solution but also of its derivatives. Furthermore, the proposed bounds may also be used to adaptively refine the train set, i.e. we start with a coarse train set and then adaptively refine the set parameter regions where the offline bounds are largest (over the tessellations/patches). This idea will be investigated in future research.

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

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

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

References

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

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

Chapter 9
ArbiLoMod: Local Solution Spaces by Random Training in Electrodynamics

Andreas Buhr, Christian Engwer, Mario Ohlberger, and Stephan Rave

Abstract The simulation method ArbiLoMod (Buhr et al., SIAM J. Sci. Comput. 2017, accepted) has the goal of providing users of Finite Element based simulation software with quick re-simulation after localized changes to the model under consideration. It generates a Reduced Order Model (ROM) for the full model without ever solving the full model. To this end, a localized variant of the Reduced Basis method is employed, solving only small localized problems in the generation of the reduced basis. The key to quick re-simulation lies in recycling most of the localized basis vectors after a localized model change. In this publication, ArbiLoMod's local training algorithm is analyzed numerically for the non-coercive problem of time harmonic Maxwell's equations in 2D, formulated in $H(\text{curl})$.

9.1 Introduction

Simulation software based on the Finite Element Method is an essential ingredient of many engineering workflows. In their pursue of design goals, engineers often simulate structures several times, applying small changes after each simulation. This results in large similarities between subsequent simulation runs. These similarities are usually not considered by simulation software. The simulation method ArbiLoMod was designed to change that and accelerate the subsequent simulation of geometries which only differ in small details. A motivating example is the design of mainboards for PCs. Improvements in the signal integrity properties of e.g. DDR memory channels is often obtained by localized changes, as depicted in Fig. 9.1.

ArbiLoMod was also designed with the available computing power in mind: Today, cloud environments are just a few clicks away and everyone can access hundreds of cores easily. However, the network connection to these cloud

A. Buhr (✉) • C. Engwer • M. Ohlberger • S. Rave
Institute for Computational and Applied Mathematics, University of Münster, Einsteinstraße 62, 48149 Münster, Germany
e-mail: andreas@andreasbuhr.de; christian.engwer@uni-muenster.de; mario.ohlberger@uni-muenster.de; stephan.rave@uni-muenster.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_9

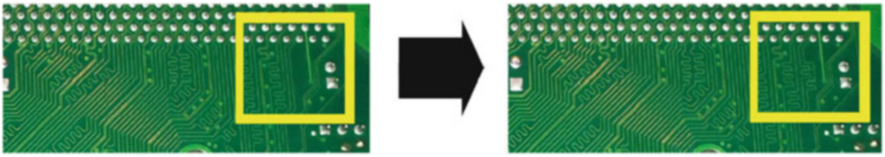


Fig. 9.1 Printed circuit board subject to local modification of conductive tracks

environments is relatively slow in comparison to the available computing power, so a method which should perform well under these circumstances must be designed to be communication avoiding.

At the core of ArbiLoMod lies a localized variant of the Reduced Basis Method. The Reduced Basis Method is a well established approach to create reduced order models (ROMs) and its application to the Maxwell's equations has been extensively investigated by many groups (see e.g. [2, 5, 9, 10, 20]). On the other hand, there are lots of methods with localized basis generation (e.g. [7, 11–13, 18, 19, 21]). However to the authors' knowledge, only little was published on the combination of both. In [4], the Reduced Basis Element Method is applied to time harmonic Maxwell's equations.

This publication evaluates ArbiLoMod's training numerically for the time harmonic Maxwell's equation. The remainder of this article is structured as follows: In the following Sect. 9.2, the problem setting is given. Section 9.3 outlines ArbiLoMod and highlights the specialties when considering inf-sup stable problems in $H(\mathrm{curl})$. Afterwards, we demonstrate ArbiLoMod's performance on a numerical example in Sect. 9.4. Finally, we conclude in Sect. 9.5.

9.2 Problem Setting

We consider Maxwell's equations [14] on the polygonal domain Ω. The material is assumed to be linear and isotropic, i.e. the electric permittivity ε and the magnetic permeability μ are scalars. On the boundary $\partial\Omega = \Gamma_R \cup \Gamma_D$ we impose Dirichlet ($E \times n = 0 \quad \text{on } \Gamma_D$) and Robin ($H \times n = \kappa(E \times n) \times n \quad \text{on } \Gamma_R$, [16, eq. (1.18)]) boundary conditions with the surface impedance parameter κ. n denotes the unit outer normal of Ω. The excitation is given by a current density $\hat{j}$.

For the time harmonic case, this results in the following weak formulation: find $u \in V := H(\mathrm{curl})$ so that

$$a(u, v; \omega) = f(v; \omega) \qquad \forall v \in V, \tag{9.1}$$

$$a(u, v; \omega) := \int_\Omega \frac{1}{\mu}(\nabla \times u) \cdot (\nabla \times \overline{v}) - \varepsilon\omega^2(u \cdot \overline{v})\mathrm{d}v + i\omega\kappa \int_{\Gamma_R} (u \times n) \cdot (\overline{v} \times n)\mathrm{d}S,$$

$$f(v; \omega) := -i\omega \int_\Omega (\hat{j} \cdot \overline{v})$$

where ω is the angular frequency. We see ω as a parametrization to this problem. We solve in a parameter domain sampled by a finite training set Ξ.

We use the inner product and energy norm given by:

$$\begin{aligned}(v,u)_V &:= \int_\Omega \frac{1}{\mu}(\nabla \times u)\cdot(\nabla \times \overline{v}) + \varepsilon\omega_{\max}^2(u\cdot\overline{v})\mathrm{d}v + \omega_{\max}\kappa\int_{\Gamma_R}(u\times n)\cdot(\overline{v}\times n)\mathrm{d}S,\\ \|u\|_V &:= \sqrt{(u,u)_V}. \end{aligned} \tag{9.2}$$

9.2.1 *Discretization*

We assume there is a non overlapping domain decomposition with subdomains Ω_i, $\Omega_i \cap \Omega_j = \emptyset$ for $i \neq j$. For simplicity, we assume it to be rectangular. The domain decomposition should cover the problem domain $\Omega \subseteq \bigcup_i \Omega_i$ but the subdomains need not resolve the domain. This is important as we want to increase or decrease the size of Ω between simulation runs without changing the domain decomposition. For example, in a printed circuit board (PCB), the metal traces are often simulated as being outside of the domain. Thus, a change of the traces leads to a change of the calculation domain.

Further we assume there is a triangulation of Ω which resolves the domain decomposition. We denote by V_h the discrete space spanned by lowest order Nedelec ansatz functions [17] on this triangulation.

9.3 ArbiLoMod for Maxwell's Equations

The main ingredients of ArbiLoMod are (1) a localizing space decomposition, (2) localized trainings for reduced local subspace generation, (3) a localized a-posteriori error estimator and (4) a localized enrichment for basis improvement. In this publication, we focus on the first two steps, which are described in the following.

9.3.1 *Space Decomposition*

Localization is performed in ArbiLoMod using a direct decomposition of the ansatz space into localized subspaces. In the 2D case with Nedelec ansatz functions, there are only volume spaces $V_{\{i\}}$ associated with the subdomains Ω_i, and interface spaces $V_{\{i,j\}}$ associated with the interfaces between Ω_i and Ω_j. The interface spaces are only associated with an interface, they are subspaces of the global function space and have support on two domains. They are not trace spaces. In higher space dimensions and/or with different ansatz functions, there may be also spaces associated with edges and nodes of the domain decomposition [1].

$$V_h = \left(\bigoplus_i V_{\{i\}} \right) \bigoplus \left(\bigoplus_{i,j} V_{\{i,j\}} \right) \tag{9.3}$$

The spaces $V_{\{i\}}$ are simply defined as the span of all ansatz functions having support only on Ω_i. With $\mathscr{B}$ denoting the set of all FE basis functions, we define:

$$V_{\{i\}} := \operatorname{span} \left\{ \psi \in \mathscr{B} \,\middle|\, \operatorname{supp}(\psi) \subseteq \overline{\Omega_i} \right\}. \tag{9.4}$$

The interface spaces $V_{\{i,j\}}$ are not simply the span of FE ansatz functions. Instead, they are defined as the span of all ansatz functions on the interface plus their extension to the adjacent subdomains. The extension is done by solving for a fixed frequency ω' with Dirichlet zero boundary conditions. The formal definition of the interface spaces is in two steps: First, we define $U_{\{i,j\}}$ as the space spanned by all ansatz function having support on both Ω_i and Ω_j:

$$U_{\{i,j\}} := \operatorname{span} \left\{ \psi \in \mathscr{B} \,\middle|\, \operatorname{supp}(\psi) \cap \Omega_i \neq \emptyset, \operatorname{supp}(\psi) \cap \Omega_j \neq \emptyset \right\}. \tag{9.5}$$

Then we define the extension operator:

$$\begin{aligned} \text{Extend} : U_{\{i,j\}} &\to V_{\{i\}} \oplus U_{\{i,j\}} \oplus V_{\{j\}}, \\ \varphi &\mapsto \varphi + \psi \\ &\quad \text{where } \psi \in V_{\{i\}} \oplus V_{\{j\}} \text{ solves} \\ &\quad a(\varphi + \psi, \phi; \omega') = 0 \qquad \forall \phi \in V_{\{i\}} \oplus V_{\{j\}}. \end{aligned} \tag{9.6}$$

We then can define the interface spaces as

$$V_{\{i,j\}} := \left\{ \text{Extend}(\varphi) \,\middle|\, \varphi \in U_{\{i,j\}} \right\}. \tag{9.7}$$

Equation (9.3) holds for this decomposition, i.e. there is a unique decomposition of every element of V_h into the localized subspaces. We define projection operators $P_{\{i\}} : V_h \to V_{\{i\}}$ and $P_{\{i,j\}} : V_h \to V_{\{i,j\}}$ by the relation

$$\varphi = \sum_i P_{\{i\}}(\varphi) + \sum_{i,j} P_{\{i,j\}}(\varphi) \qquad \forall \varphi \in V_h. \tag{9.8}$$

9.3.2 Training

Having defined the localized spaces, we create reduced localized subspaces $\widetilde{V}_{\{i\}} \subseteq V_{\{i\}}$ and $\widetilde{V}_{\{i,j\}} \subseteq V_{\{i,j\}}$ by a localized training procedure. The training is inspired by the “Empirical Port Reduction” introduced by Eftang et al. [8]. Its main four steps are:

1. solve the problem (9.1) on a small training domain around the space in question with zero boundary values for all parameters in the training set Ξ,
2. solve the homogeneous equation repeatedly on a small training domain around the space in question with random boundary values for all parameters in Ξ,
3. apply the space decomposition to all computed local solutions to obtain the part belonging to the space in question and
4. use a greedy procedure to create a space approximating this set.

For further details, we refer to [1]. The small training domain for an interface space consists of the six subdomains around that interface. The small training domain for a volume space consists of nine subdomains: the subdomain in question and the eight subdomains surrounding it.

While the “Empirical Port Reduction” in [8] generates an interface space and requires ports which do not intersect, this training can be used for both interface and volume spaces. It can handle touching ports and can thus be applied to a standard domain decomposition.

9.3.3 Reduced Model

In these first experiments the reduced global problem is obtained by a simple Galerkin projection onto the direct sum of all reduced local subspaces. The global solution space is

$$\widetilde{V}_h := \left(\bigoplus_i \widetilde{V}_{\{i\}} \right) \bigoplus \left(\bigoplus_{i,j} \widetilde{V}_{\{i,j\}} \right) . \tag{9.9}$$

And the reduced problem reads: find $\widetilde{u} \in \widetilde{V}_h$ such that

$$a(\widetilde{u}, v; \mu) = f(v) \qquad \forall v \in \widetilde{V}_h \, . \tag{9.10}$$

9.4 Numerical Example

The numerical experiments are performed with pyMOR [15]. The source code used to reproduce the results in this publication is provided alongside with this publication and can be downloaded at http://www.arbilomod.org/morepas2015.tgz. See the README file therein for installation instructions.

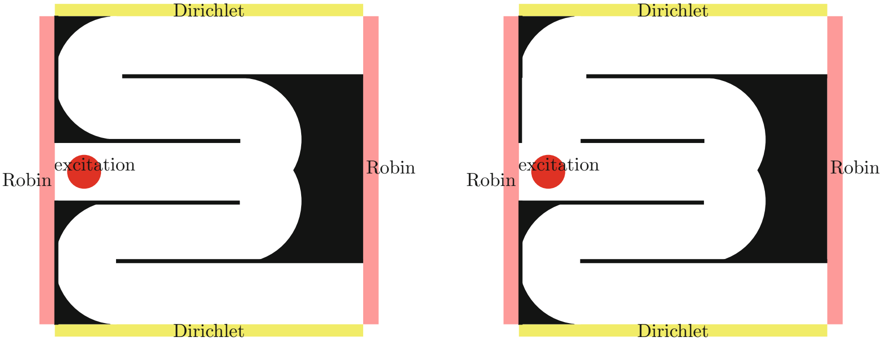


Fig. 9.2 Geometries simulated. Black area is not part of the domain and treated as Dirichlet zero boundary. Note the change is topology changing

9.4.1 Geometry Simulated

We simulate the unit square $(0, 1) \times (0, 1)$ with robin boundary conditions at $\Gamma_R := 0 \times (0, 1) \cup 1 \times (0, 1)$ and Dirichlet zero boundary conditions at $\Gamma_D := (0, 1) \times 0 \cup (0, 1) \times 1$. The surface impedance parameter κ is chosen as the impedance of free space, $\kappa = 1/376.73$ Ohm. We introduce some structure by inserting perfect electric conductors (PEC) into the domain, see Fig. 9.2. The PEC is modeled as Dirichlet zero boundary condition. Note that it is slightly asymmetric intentionally, to produce more interesting behavior. The mesh does not resolve the geometry. Rather, we use a structured mesh and remove all degrees of freedom which are associated with an edge whose center is inside of the PEC structure. The structured mesh consists of 100 times 100 squares, each of which is divided into four triangles. With each edge, one degree of freedom is associated, which results in 60,200 degrees of freedom, some of which are "disabled" because they are in PEC or on a Dirichlet boundary. The parameter domain is the range from 10 MHz to 1 GHz. For the training set Ξ, we use 100 equidistant points in this range, including the endpoints. To simulate an "arbitrary local modification", the part of the PEC within $(0.01, 0.2) \times (0.58, 0.80)$ is removed and the simulation domain is enlarged.

The excitation is a current

$$j(x, y) := \exp\left(-\frac{(x - 0.1)^2 + (y - 0.5)^2}{1.25 \cdot 10^{-3}}\right) \cdot \mathrm{e}_y \tag{9.11}$$

To get an impression of the solutions, some example solutions are plotted in Fig. 9.3.

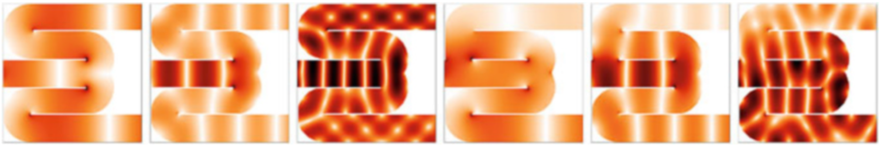


Fig. 9.3 Example solutions for f=186 MHz, f=561 MHz and f=1 GHz for the first and second geometry. Plotted is $|\mathrm{Re}(E)|$. Script: maxwell_create_solutions.py

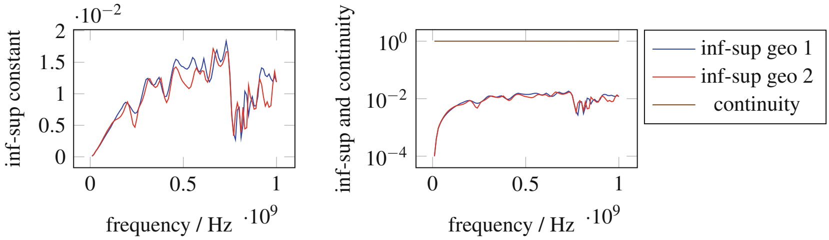


Fig. 9.4 inf-sup and continuity constant of bilinear form. Linear and logarithmic. Script: maxwell_calculate_infsup.py

9.4.2 Global Properties of Example

Before analyzing the behavior of the localized model reduction, we discuss some properties of the full model. For its stability, its continuity constant γ and reduced inf-sup constants $\widetilde{\beta}$ are the primary concern. They guarantee existence and uniqueness of the solution and their quotient enters the best-approximation inequality

$$\|u - \widetilde{u}\| \leq \left(1 + \frac{\gamma}{\widetilde{\widetilde{\beta}}}\right) \inf_{v \in \widetilde{V_h}} \|u - v\| \tag{9.12}$$

where u is the solution in V_h. Due to the construction of the norm, the continuity constant cannot be larger than one, and numerics indicate that it is usually one (Fig. 9.4). The inf-sup constant approaches zero when the frequency goes to zero. This is the well known low frequency instability of this formulation. There are remedies to this problem, but they are not considered here. The order of magnitude of the inf-sup constant is around 10^{-2}: Due to the Robin boundaries, the problem is stable. With Dirichlet boundaries only, the inf-sup constant would drop to zero at several frequencies. There are two drops in the inf-sup constant at ca. 770 MHz and 810 MHz. These correspond to resonances in the structure which arise when half a wavelength is the width of a channel ($\lambda/2 \approx 1/5$).

The most important question for the applicability of any reduced basis method is: is the system reducible at all, i.e. can the solution manifold be approximated with a low dimensional solution space? The best possible answer to this question is the Kolmogorov n-width. We measured the approximation error when approximating

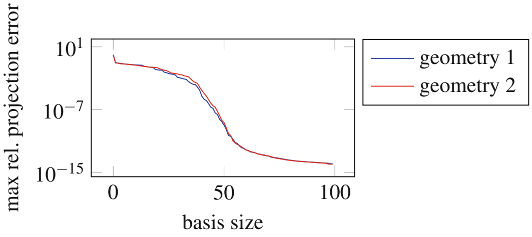


Fig. 9.5 Error when approximating the solution set for all $f \in \Xi$ with an n-dimensional basis obtained by greedy approximation of this set. This is an upper bound for the Kolmogorov n-width. Script: maxwell_global_n_width.py

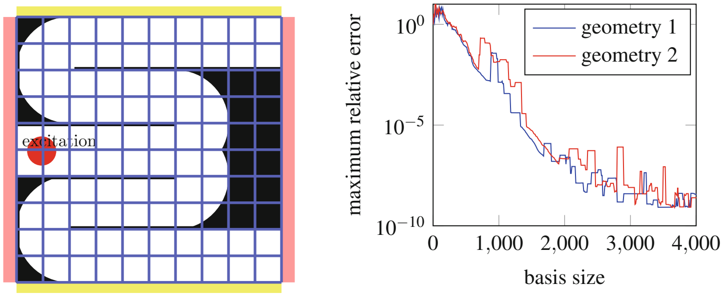


Fig. 9.6 *Left*: Domain decomposition used. *Right*: Maximum error when solving with a localized basis, generated by global solves. Script maxwell_local_n_width.py

the solution manifold with a basis generated by a greedy algorithm. The approximation is done by orthogonal projection onto the basis. This error is an upper bound to the Kolmogorov n-width. Already with a basis size of 38, a relative error of 10^{-4} can be achieved, see Fig. 9.5. So this problem is well suited for reduced basis methods.

9.4.3 Properties of Localized Spaces

The next question is: how much do we lose by localization? Using basis vectors with limited support, one needs a larger total number of basis functions. To quantify this, we compare the errors with global approximation from the previous section with the error obtained when solving with a localized basis, using the best localized basis we can generate. We use a 10 x 10 domain decomposition (see Fig. 9.6 left) and the space decomposition introduced in Sect. 9.3.1. To construct the best possible basis, we solve the full problem for all parameters in the training set. For each local subspace, we apply the corresponding projection operator $P_{\{i\}}$ / $P_{\{i,j\}}$ to all global solutions and subsequently generate a basis for these local parts of global solutions using a greedy procedure. The error when solving in the resulting reduced space is depicted in Fig. 9.6, right. Much more basis vectors are needed, compared to the

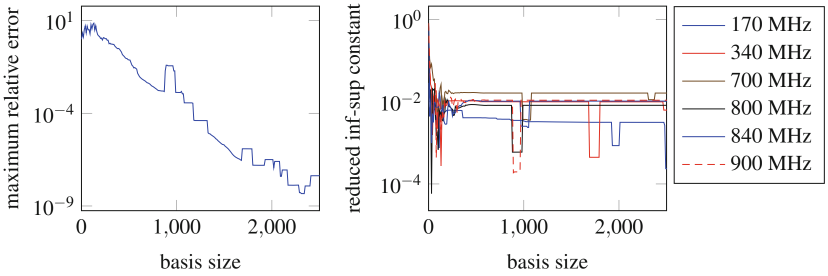


Fig. 9.7 Comparison of maximum error over all frequencies with inf-sup constant of reduced system at selected frequencies for geometry 1. Basis generated by global solves. Increased error and reduced inf-sup constant around basis size of 900. Script: maxwell_infsup_during_reduction.py

global reduced basis approach. However the reduction in comparison to the full model (60,200 dofs) is still significant and in contrast to standard reduced basis methods, the reduced system matrix is not dense but block-sparse. For a relative error of 10^{-4}, 1080 basis vectors are necessary.

In Fig. 9.6 the error is observed to jump occasionally. This is due to the instability of a Galerkin projection of an inf-sup stable problem. While coercivity is retained during Galerkin projection, inf-sup stability is not. While the inf-sup constant of the reduced system is observed to be the same as the inf-sup constant of the full system most of the time, sometimes it drops. This is depicted in Fig. 9.7. For a stable reduction, a different test space is necessary. However, the application of the known approaches such as [3, 6] to the localized setting is not straightforward. The development of stable test spaces in the localized setting is beyond the scope of this publication.

9.4.4 Properties of Training

Local basis vectors should be generated using the localized training described in Sect. 9.3.2 and in [1]. To judge on the quality of these basis vectors, we compare the error obtained using these basis vectors with the error obtained with local basis vectors generated by global solves. The local basis vectors generated by global solves are the reference: These are the best localized basis we can generate. The results for both geometries are depicted in Fig. 9.8. While the error decreases more slowly, we still have reasonable basis sizes with training. For a relative error of 10^{-4}, 1280 basis vectors are necessary for geometry 1 and 1380 are necessary for geometry 2.

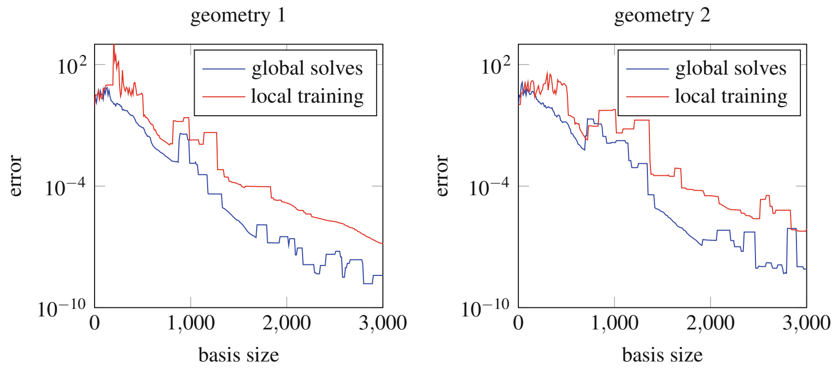


Fig. 9.8 Maximum error over all frequencies for both geometries. Basis generated by global solves vs. basis generated by local training. Script: maxwell_training_benchmark.py

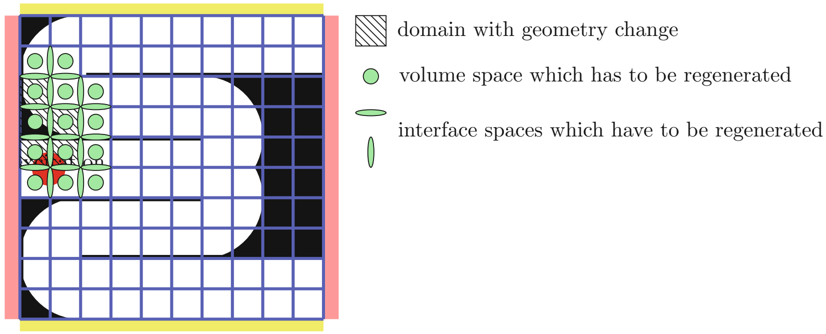

domain with geometry change
volume space which has to be regenerated
interface spaces which have to be regenerated

Fig. 9.9 Impact of geometry change: 5 domains contain changes, 14 domain spaces and 20 interface spaces have to be regenerated

9.4.5 Application to Local Geometry Change

If we work with a relative error of 5%, a basis of size 650 is sufficient for the first geometry and size 675 for the second. After the geometry change, the local reduced spaces which have no change in their training domain can be reused. Instead of solving the full system with 60,200 degrees of freedom, the following effort is necessary per frequency point (see also Fig. 9.9). Because the runtime is dominated by matrix factorizations, we focus on these.

- 14 factorizations of local problems with 5340 dofs (volume training)
- 20 factorizations of local problems with 3550 dofs (interface training)
- 1 factorization of global reduced problem with 675 dofs (global solve)

The error between the reduced solution and the full solution in this case is 4.3%. Script to reproduce: experiment_maxwell_geochange.py. The spacial distribution of basis sizes is depicted in Fig. 9.10.

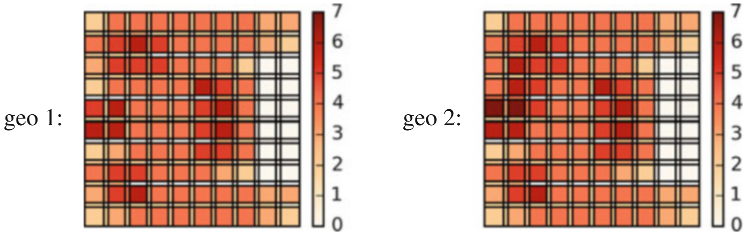


Fig. 9.10 Basis size distribution. Script: postprocessing_draw_basis_sizes_maxwell_geochange.py

9.5 Conclusion

ArbiLoMod was applied to the non-coercive problem of 2D Maxwell's equations in $H(\mathrm{curl})$. Its localized training generates a basis of good quality. A reduced model with little error for the full problem can be generated using only local solves, which can easily be parallelized. After localized changes to the model, only in the changed region the localized bases have to be regenerated. All other bases can be reused, which results in large computational savings compared to a simulation from scratch. The amount of savings is very dependent of the model and the changes which are made. A thorough analysis of the computational savings is subject to future work, as is the adaptation of ArbiLoMod's localized a-posteriori error estimator and online enrichment to this problem as well as the instability of the Galerkin projection.

Acknowledgements Andreas Buhr was supported by CST—Computer Simulation Technology AG. Stephan Rave was supported by the German Federal Ministry of Education and Research (BMBF) under contract number 05M13PMA.

References

1. Buhr, A., Engwer, C., Ohlberger, M., Rave, S.: ArbiLoMod, a simulation technique designed for arbitrary local modifications. SIAM J. Sci. Comput. (2017). Accepted
2. Burgard, S., Sommer, A., Farle, O., Dyczij-Edlinger, R.: Reduced-order models of finite-element systems featuring shape and material parameters. Electromagnetics **34**(3–4), 143–160 (2014)
3. Carlberg, K., Bou-Mosleh, C., Farhat, C.: Efficient non-linear model reduction via a least-squares Petrov–Galerkin projection and compressive tensor approximations. Int. J. Numer. Methods Eng. **86**(2), 155–181 (2011)
4. Chen, Y., Hesthaven, J.S., Maday, Y.: A seamless reduced basis element method for 2D Maxwell's problem: an introduction. Spectral and High Order Methods for Partial Differential Equations: Selected papers from the ICOSAHOM '09 Conference, June 22–26, Trondheim, Norway, pp. 141–152. Springer, Berlin (2011)
5. Chen, Y., Hesthaven, J.S., Maday, Y., Rodríguez, J.: Certified reduced basis methods and output bounds for the harmonic Maxwell's equations. SIAM J. Sci. Comput. **32**(2), 970–996 (2010)

6. Dahmen, W., Plesken, C., Welper,G.: Double greedy algorithms: reduced basis methods for transport dominated problems. ESAIM: Math. Model. Numer. Anal. **48**(03), 623–663 (2014)
7. Efendiev, Y., Galvis, J., Hou, T.Y.: Generalized multiscale finite element methods (GMsFEM). J. Comput. Phys. **251**, 116–135 (2013)
8. Eftang, J.L., Patera, A.T.: Port reduction in parametrized component static condensation: approximation and a posteriori error estimation. Int. J. Numer. Methods Eng. **96**(5), 269–302 (2013)
9. Fares, M., Hesthaven, J.S., Maday, Y., Stamm, B.: The reduced basis method for the electric field integral equation. J. Comput. Phys. **230**(14), 5532–5555 (2011)
10. Hess, M.W., Benner, P.: Fast evaluation of time harmonic Maxwell's equations using the reduced basis method. IEEE Trans. Microw. Theory Tech. **61**(6), 2265–2274 (2013)
11. Iapichino, L., Quarteroni, A., Rozza,G.: A reduced basis hybrid method for the coupling of parametrized domains represented by fluidic networks. Comput. Methods Appl. Mech. Eng. **221–222**, 63–82 (2012)
12. Maday, Y., Rønquist, E.M.: A reduced-basis element method. J. Sci. Comput. **17**(1/4), 447–459 (2002)
13. Maier, I., Haasdonk, B.: A Dirichlet-Neumann reduced basis method for homogeneous domain decomposition problems. Appl. Numer. Math. **78**, 31–48 (2014)
14. Maxwell, J.C.: On physical lines of force. Lond. Edinb. Dublin Philos. Mag. J. Sci. **21**(139), 161–175 (1861)
15. Milk, R., Rave, S., Schindler, F.: pyMOR-generic algorithms and interfaces for model order reduction. SIAM J. Sci. Comput. **38**(5), S194–S216 (2016). doi:10.1137/15M1026614, https://doi.org/10.1137/15M1026614
16. Monk, P.: Finite Element Methods for Maxwell's Equations Oxford University Press, Oxford (2003)
17. Nedelec, J.-C.: Mixed finite elements in r^3. Numer. Math. **35**(3), 315–341 (1980)
18. Ohlberger, M., Schindler, F.: Error control for the localized reduced basis multi-scale method with adaptive on-line enrichment. SIAM J. Sci. Comput. **37**(6), A2865–A2895 (2015)
19. Phuong Huynh, D.B., Knezevic, D.J., Patera, A.T.: A static condensation reduced basis element method : approximation and a posteriori error estimation. ESAIM: Math. Model. Numer. Anal. **47**(1), 213–251 (2012)
20. Pomplun, J., Schmidt, F.: Accelerated a posteriori error estimation for the reduced basis method with application to 3d electromagnetic scattering problems. SIAM J. Sci. Comput. **32**(2), 498–520 (2010)
21. Strouboulis, T., Babuška, I., Copps, K.: The design and analysis of the generalized finite element method. Comput. Methods Appl. Mech. Eng. **181**(1), 43–69 (2000)

Chapter 10
Reduced-Order Semi-Implicit Schemes for Fluid-Structure Interaction Problems

Francesco Ballarin, Gianluigi Rozza, and Yvon Maday

Abstract POD–Galerkin reduced-order models (ROMs) for fluid-structure interaction problems (incompressible fluid and thin structure) are proposed in this paper. Both the high-fidelity and reduced-order methods are based on a Chorin-Temam operator-splitting approach. Two different reduced-order methods are proposed, which differ on velocity continuity condition, imposed weakly or strongly, respectively. The resulting ROMs are tested and compared on a representative haemodynamics test case characterized by wave propagation, in order to assess the capabilities of the proposed strategies.

10.1 Introduction

Several applications are characterized by multi-physics phenomena, such as the interaction between an incompressible fluid and a compressible structure. The capability to perform real-time multi-physics simulations could greatly increase the applicability of computational methods in applied sciences and engineering. To reach this goal, reduced-order modelling techniques are applied in this paper. We refer the interested reader to [1, 6, 9, 19] for some representative previous approaches to the reduction of fluid-structure interaction problems, arising in aeroelasticity [1] or haemodynamics [6, 9, 19]. The reduction proposed in the current work is based on a POD–Galerkin approach. A difference with our previous work [6], where a reduced-order monolithic approach has been proposed, is related to the use of a partitioned reduced-order model, based on the semi-implicit operator-spitting approach originally employed in [11] for the high-fidelity method.

F. Ballarin (✉) • G. Rozza
Mathematics Area, mathLab, SISSA, via Bonomea 265, I-34136, Trieste, Italy
e-mail: francesco.ballarin@sissa.it; gianluigi.rozza@sissa.it

Y. Maday
Sorbonne Universités, UPMC Université Paris 06 and CNRS UMR 7598, Laboratoire Jacques-Louis Lions, F-75005, Paris, France

Division of Applied Mathematics, Brown University, Providence, RI, USA
e-mail: maday@ann.jussieu.fr

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_10

Extension to other methods (e.g. [4, 16]) is possible and object of forthcoming work. The formulation of the FSI problem is summarized in Sect. 10.2, and its high-fidelity discretization is reported in Sect. 10.3. Two reduced-order models are proposed in Sect. 10.4, and compared by means of a numerical test case in Sect. 10.5. Conclusions and perspectives are summarized in the final section of the paper.

10.2 Problem Formulation

In this section the formulation of the fluid-structure interaction (FSI) model problem is summarized. Let us consider the bidimensional fluid domain $\Omega = [0, L] \times [0, h_f]$. Its boundary is composed of a compliant wall $\Sigma = [0, L] \times \{h_f\}$ (top), fluid inlet section $\Gamma_{in} = \{0\} \times [0, h_f]$ (left) and fluid outlet section $\Gamma_{out} = \{L\} \times [0, h_f]$ (right), and a wall $\Gamma_{sym} = [0, L] \times \{0\}$ (bottom). For the sake of simplicity we assume low Reynolds numbers for the fluid problem and infinitesimal displacements for the compliant wall. Thus, in the following, we will consider *unsteady Stokes* equations on a *fixed* domain Ω for the fluid and a linear structural model for the compliant wall. In particular, we will further assume that the structure undergoes negligible horizontal displacements, so that the structural equations of the compliant wall can be described by a *generalized string model* [21, 22].

The coupled fluid-structure interaction problem is therefore: for all $t \in (0, T]$, find fluid velocity $\boldsymbol{u}(t) : \Omega \to \mathbb{R}^2$, fluid pressure $p(t) : \Omega \to \mathbb{R}$ and structure displacement $\eta(t) : \Sigma \to \mathbb{R}$ such that

$$\begin{cases} \rho_f \partial_t \boldsymbol{u} - \operatorname{div}(\boldsymbol{\sigma}(\boldsymbol{u}, p)) = \boldsymbol{0} & \text{in } \Omega \times (0, T], \\ \operatorname{div} \boldsymbol{u} = 0 & \text{in } \Omega \times (0, T], \\ \boldsymbol{u} = \partial_t \eta \, \boldsymbol{n} & \text{on } \Sigma \times (0, T], \\ \rho_s h_s \partial_{tt} \eta - c_1 \partial_{xx} \eta + c_0 \eta = -\boldsymbol{\sigma}(\boldsymbol{u}, p) \boldsymbol{n} \cdot \boldsymbol{n} & \text{on } \Sigma \times (0, T]. \end{cases} \tag{10.1}$$

Equations $(10.1)_1$–$(10.1)_2$ formulate the Stokes problem on the fixed fluid domain Ω, having defined the fluid Cauchy stress tensor $\boldsymbol{\sigma}(\boldsymbol{u}, p) := -p\boldsymbol{I} + 2\mu_f \boldsymbol{\varepsilon}(\boldsymbol{u})$, $\boldsymbol{\varepsilon}(\boldsymbol{u}) := \frac{1}{2}(\nabla \boldsymbol{u} + \nabla^T \boldsymbol{u})$, while $(10.1)_4$ is the equation of the structural motion of the compliant wall. Continuity of the velocity on the interface is guaranteed by $(10.1)_3$. The FSI system is completed by suitable initial and boundary conditions. In this paper we assume resting conditions at $t = 0$ and the following fluid boundary conditions on $\partial\Omega \setminus \Sigma$:

$$\begin{cases} \boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n} = -p_{in}(t)\boldsymbol{n} & \text{on } \Gamma_{in} \times (0, T], \\ \boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n} = -p_{out}(t)\boldsymbol{n} & \text{on } \Gamma_{out} \times (0, T], \\ \boldsymbol{u} \cdot \boldsymbol{n} = 0, \quad \boldsymbol{\sigma}(\boldsymbol{u}, p)\boldsymbol{n} \cdot \boldsymbol{\tau} = 0 & \text{on } \Gamma_{sym} \times (0, T], \end{cases} \tag{10.2}$$

and the following structure boundary condition on $\partial\Sigma$:

$$\eta = 0 \quad \text{on } \partial\Sigma \times (0, T]. \tag{10.3}$$

Equations $(10.2)_1$ and $(10.2)_2$ prescribe pressure on inlet and outlet section, respectively; Eq. $(10.2)_3$ is a symmetry condition, arising from the consideration of this simplified 2D problem as a section of a 3D cylindrical configuration. Here $\boldsymbol{n}$ and $\boldsymbol{\tau}$ denote the outer unit normal to Ω and tangential vector to Σ, respectively. Finally, (10.3) prescribes a clamped wall near both the inlet and outlet section of the fluid. The value of constitutive parameters ρ_f (fluid density), ρ_s (structure density), μ_f (fluid viscosity), c_1 and c_0 (structure constitutive parameters), L (domain width), h_f (fluid height), h_s (structure thickness) will be further specified in Sect. 10.5, as well as choices of $p_{in}(t)$, $p_{out}(t)$ and T for which the FSI system (10.1)–(10.3) is characterized by propagation of a pressure wave.

10.3 High-Fidelity Formulation: Semi-Implicit Scheme

In this section we summarize the high-fidelity discretization of the FSI system (10.1)–(10.3). An operator splitting approach, based on a Chorin-Temam projection scheme, is pursued. In particular, Robin-Neumann iterations are carried out in order to enhance the stability of the resulting algorithm.

10.3.1 A Projection-Based Semi-Implicit Coupling Scheme

We employ in this work a projection-based semi-implicit scheme, as proposed in [2, 11]. The fluid equations are discretized in time using the Chorin-Temam projection scheme [23]. Thus, denoting by Δt the time step length, $D_t f^{k+1} := \frac{f^{k+1}-f^k}{\Delta t}$ the first backward difference approximation of the time derivative of $f(t^{n+1})$ and $D_{tt} f^{k+1} = D_t(D_t f^{k+1})$, we consider the following semi-implicit time discretization of (10.1)–(10.3): for any $k = 1, \dots, K = T/\Delta t$

1. Explicit step (fluid viscous part): find[1] $\widetilde{\boldsymbol{u}}^{k+1} : \Omega \to \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f \frac{\widetilde{\boldsymbol{u}}^{k+1} - \widetilde{\boldsymbol{u}}^k}{\Delta t} - 2\mu_f \operatorname{div} \boldsymbol{\varepsilon}(\widetilde{\boldsymbol{u}}^{k+1}) = -\nabla p^k & \text{in } \Omega, \\ \widetilde{\boldsymbol{u}}^{k+1} = D_t \eta^k \boldsymbol{n} & \text{on } \Sigma. \end{cases} \tag{10.4}$$

[1] For the sake of an easier comparison with Remark 1 we employ here the $\widetilde{\cdot}$ notation for the velocity. However, since we will always employ the pressure Poisson formulation, the $\widetilde{\cdot}$ will be dropped in the next sections.

2. Implicit step:

   2.1. Fluid projection substep: find $p^{k+1} : \Omega \to \mathbb{R}$ such that:

$$\begin{cases} -\operatorname{div}(\nabla p^{k+1}) = -\frac{\rho_f}{\Delta t} \operatorname{div} \widetilde{\boldsymbol{u}}^{k+1} & \text{in } \Omega, \\ \frac{\partial}{\partial \boldsymbol{n}} p^{k+1} = -\rho_f D_{tt} \eta^{k+1} & \text{on } \Sigma. \end{cases} \tag{10.5}$$

   2.2. Structure substep: find $\eta^{k+1} : \Sigma \to \mathbb{R}$ such that:

$$\rho_s h_s D_{tt} \eta^{k+1} - c_1 \partial_{xx} \eta^{k+1} + c_0 \eta^{k+1} = -\boldsymbol{\sigma}(\widetilde{\boldsymbol{u}}^{k+1}, p^{k+1}) \boldsymbol{n} \cdot \boldsymbol{n} \quad \text{on } \Sigma. \tag{10.6}$$

The implicit step couples pressure stresses to the structure, and it is iterated until convergence.

Remark 1 In place of step 1 and 2.1 (pressure Poisson formulation) one could also consider the following pressure Darcy formulation:

I. Explicit step (fluid viscous part): find $\widetilde{\boldsymbol{u}}^{k+1} : \Omega \to \mathbb{R}^2$ such that:

$$\begin{cases} \rho_f \frac{\widetilde{\boldsymbol{u}}^{k+1} - \boldsymbol{u}^k}{\Delta t} - 2\mu_f \operatorname{div} \boldsymbol{\varepsilon}(\widetilde{\boldsymbol{u}}^{k+1}) = \boldsymbol{0} & \text{in } \Omega, \\ \widetilde{\boldsymbol{u}}^{k+1} = D_t \eta^k \boldsymbol{n} & \text{on } \Sigma. \end{cases}$$

II. Implicit step:

   II.1. Fluid projection substep: find $\boldsymbol{u}^{k+1} : \Omega \to \mathbb{R}^2$ and $p^{k+1} : \Omega \to \mathbb{R}$ such that:

$$\begin{cases} \rho_f \frac{\boldsymbol{u}^{k+1} - \widetilde{\boldsymbol{u}}^{k+1}}{\Delta t} + \nabla p^{k+1} = \boldsymbol{0} & \text{in } \Omega, \\ \boldsymbol{u}^{k+1} \cdot \boldsymbol{n} = D_t \eta^{k+1} & \text{on } \Sigma. \end{cases}$$

   II.2. Structure substep: find $\eta^{k+1} : \Sigma \to \mathbb{R}$ such that:

$$\rho_s h_s D_{tt} \eta^{k+1} - c_1 \partial_{xx} \eta^{k+1} + c_0 \eta^{k+1} = -\boldsymbol{\sigma}(\widetilde{\boldsymbol{u}}^{k+1}, p^{k+1}) \boldsymbol{n} \cdot \boldsymbol{n} \quad \text{on } \Sigma.$$

For the sake of a more efficient reduced-order model (see Sect. 10.4) it is convenient to consider the pressure Poisson formulation (steps 1 and 2.1) rather than the pressure Darcy formulation (step I and II.1), because the latter would require a larger system (comprised of both velocity and pressure) at step II.1.

Finally, in order to enhance the stability of the projection method we employ a Robin-Neumann coupling, as proposed in [2]. See also [3, 12] for related topics. Thus, we replace $(10.5)_2$ with

$$\frac{\partial}{\partial \boldsymbol{n}} p^{k+1} + \alpha_{Rob} p^{k+1} = -\rho_f D_{tt} \eta^{k+1} + \alpha_{Rob} p^{k,*} \quad \text{on } \Sigma. \tag{10.7}$$

being $\alpha_{Rob} > 0$ and $p^{k,*}$ an extrapolation of the pressure (which will be defined in Sect. 10.3.2). In particular, following [10, 13], we choose $\alpha_{Rob} := \frac{\rho_f}{\rho_s h_s}$. We remark that, due to the simplifying assumptions of this problem (linear structural model, fixed domain), the implicit step could have been solved in one shot, since it defines a linear system in (p^{k+1}, η^{k+1}). We still keep the Robin-Neumann coupling in this case in order to assess the capabilities of such procedure in a reduced-order setting, since it will be required in more general nonlinear problems.

10.3.2 Space Discretization of the High-Fidelity Formulation

Denote by $V = [H^1(\Omega)]^2$ the fluid velocity space (endowed with the H^1 seminorm), by $Q = L^2(\Omega)$ the fluid pressure space (endowed with the L^2 norm), and by $E = H^1(\Sigma)$ the structure displacement space (endowed with the H^1 seminorm). After having obtained a weak formulation of the semi-implicit formulation, we consider a finite element (FE) discretization for steps 1, 2.1 and 2.2. Second order Lagrange FE are employed for fluid velocity (step 1) and structural displacement (step 2.2), resulting in FE spaces $V_h \subset V$ and $E_h \subset E$, respectively, while fluid pressure is discretized by first order Lagrange FE, $Q_h \subset Q$. Thus, the corresponding Galerkin-FE formulation reads: for any $k = 1, \ldots, K$

1_h. Explicit step (fluid viscous part): find $\boldsymbol{u}_h^{k+1} \in V_h$ such that:

$$\int_\Omega \frac{\rho_f}{\Delta t}\boldsymbol{u}_h^{k+1} \cdot \boldsymbol{v}_h \, d\boldsymbol{x} + \int_\Omega 2\mu_f \boldsymbol{\varepsilon}(\boldsymbol{u}_h^{k+1}) : \nabla \boldsymbol{v}_h \, d\boldsymbol{x} = \int_\Omega \frac{\rho_f}{\Delta t}\boldsymbol{u}_h^{k} \cdot \boldsymbol{v}_h \, d\boldsymbol{x} - \int_\Omega \nabla p_h^k \cdot \boldsymbol{v}_h \, d\boldsymbol{x} \tag{10.8}$$

for all $\boldsymbol{v}_h \in V_h$, subject to the coupling condition

$$\boldsymbol{u}_h^{k+1} = D_t \eta_h^k \boldsymbol{n} \quad \text{on } \Sigma \times (0, T], \tag{10.9}$$

and to the boundary condition $\boldsymbol{u}_h^{k+1} \cdot \boldsymbol{n} = 0$ on $\Gamma_{sym} \times (0, T]$.

2_h. Implicit step: for any $j = 0, \ldots,$ until convergence:

2.1_h. Fluid projection substep, with Robin boundary conditions: find $p_h^{k+1,j+1} \in Q_h$ such that:

$$\begin{aligned}\int_\Omega \nabla p_h^{k+1,j+1} \cdot \nabla q_h \, d\boldsymbol{x} + \int_\Sigma \alpha_{Rob} \, p_h^{k+1,j+1} \, q_h \, ds = \\ - \int_\Omega \frac{\rho_f}{\Delta t} \operatorname{div} \boldsymbol{u}_h^{k+1} \, q_h \, d\boldsymbol{x} - \int_\Sigma \rho_f D_{tt} \eta_h^{k+1,j} \, q_h \, ds \\ + \int_\Sigma \alpha_{Rob} \, p_h^{k+1,j} \, q_h \, ds\end{aligned}$$

for all $q_h \in Q_h$, subject to the boundary conditions $p_h^{k+1,j+1} = p_{in}(t)$ on $\Gamma_{in} \times (0,T]$ and $p_h^{k+1,j+1} = p_{out}(t)$ on $\Gamma_{out} \times (0,T]$. Here the value $p_h^{k+1,j}$ has been chosen as pressure extrapolation for the Robin-Neumann coupling.

2.2$_h$. Structure substep: find $\eta_h^{k+1,j+1} \in E_h$ such that:

$$\begin{aligned}\int_\Sigma \frac{\rho_s h_s}{\Delta t^2}\,\eta_h^{k+1,j+1}\,\zeta_h\,ds &+ \int_\Sigma c_1 \partial_x \eta_h^{k+1,j+1}\,\partial_x\zeta_h\,ds \\ &+ \int_\Sigma c_0 \eta_h^{k+1,j+1}\,\zeta_h\,ds = \int_\Sigma \frac{\rho_s h_s}{\Delta t^2}\eta_h^k\,\zeta_h\,ds \\ &+ \int_\Sigma \frac{\rho_s h_s}{\Delta t} D_t \eta_h^k\,\zeta_h\,ds \\ &- \int_\Sigma \boldsymbol{\sigma}(\boldsymbol{u}^{k+1}, p^{k+1,j+1})\boldsymbol{n}\cdot\zeta_h \boldsymbol{n}\,ds\end{aligned} \tag{10.10}$$

for all $\zeta_h \in E_h$, subject to the boundary conditions $\eta_h^{k+1,j+1} = 0$ on $\partial\Sigma$.

The coupling condition (10.9) is imposed strongly. We will further comment in Sect. 10.4 on the imposition of this condition at the reduced-order level. A relative error on the increments is chosen as stopping criterion for step 2_h, that is the implicit step is repeated until

$$\min\left\{ \frac{\left\| p_h^{k+1,j+1} - p_h^{k+1,j} \right\|_Q}{\left\| p_h^{k+1,j+1} \right\|_Q}, \frac{\left\| \eta_h^{k+1,j+1} - \eta_h^{k+1,j} \right\|_E}{\left\| \eta_h^{k+1,j+1} \right\|_E} \right\} < \mathrm{tol},$$

for some prescribed tolerance tol. The solution $(p_h^{k+1,j^*}, \eta_h^{k+1,j^*})$ at the iteration j^* such that convergence is achieved is then denoted by $(p_h^{k+1}, \eta_h^{k+1})$.

10.4 Reduced-Order Formulation: POD–Galerkin Semi-Implicit Scheme

In this section we propose two Proper Orthogonal Decomposition (POD)–Galerkin semi-implicit reduced-order models (ROMs) for FSI system (10.1)–(10.3). The first ROM (FSI ROM 1) is built starting directly from steps 1_h, 2.1_h and 2.2_h, and performing a Galerkin projection. Special treatment will be devoted to the imposition of the coupling condition (10.9); unfortunately, this requires enlarging the reduced-order systems. The second ROM (FSI ROM 2) will exploit a simple change of variable for the fluid velocity to bypass this issue. In both cases, an offline-online computational decoupling is sought [26].

10.4.1 FSI ROM 1 Approach

10.4.1.1 Offline Stage

During the offline stage, the solution of the high-fidelity problem 1_h, 2.1_h and 2.2_h is computed. We then consider the following snapshot matrices

$$S_{\boldsymbol{u}} = [\underline{\mathbf{u}}_h^1|\dots|\underline{\mathbf{u}}_h^K] \in \mathbb{R}^{N_h^{\boldsymbol{u}}\times K},$$

$$S_p = [\underline{\mathbf{p}}_h^1|\dots|\underline{\mathbf{p}}_h^K] \in \mathbb{R}^{N_h^p\times K},$$

$$S_\eta = [\underline{\boldsymbol{\eta}}_h^1|\dots|\underline{\boldsymbol{\eta}}_h^K] \in \mathbb{R}^{N_h^\eta\times K},$$

where we denote with the underlined notation the vector of FE degrees of freedom corresponding to each solution. Here $N_h^{\boldsymbol{u}} = \dim(V_h)$, $N_h^p = \dim(Q_h)$ and $N_h^\eta = \dim(E_h)$. Then, we carry out a proper orthogonal decomposition of each snapshot matrix; the method of snapshots is used, and the snapshots are weighted with the inner product associated to their functional space. Then, the first $N^{\boldsymbol{u}}$, N^p and N^η (respectively) left singular vectors, denoted by $\{\boldsymbol{\varphi}_i\}_{i=1}^{N^{\boldsymbol{u}}}$, $\{\psi_j\}_{j=1}^{N^p}$ and $\{\phi_l\}_{l=1}^{N^\eta}$ (resp.), are then chosen as basis functions for the reduced spaces $V_N^{(1)}$, $Q_N^{(1)}$ and $E_N^{(1)}$ (resp.), i.e.

$$V_N^{(1)} = \text{span}(\{\boldsymbol{\varphi}_i\}_{i=1}^{N^{\boldsymbol{u}}}), \quad Q_N^{(1)} = \text{span}(\{\psi_j\}_{j=1}^{N^p}), \quad E_N^{(1)} = \text{span}(\{\phi_l\}_{l=1}^{N^\eta}).$$

10.4.1.2 On the Imposition of Coupling Condition (10.9)

The major drawback of this approach is related to the fact that the reduced spaces $V_N^{(1)}$ and $E_N^{(1)}$ do not guarantee, in general, that the coupling conditions (10.9) holds. To this end, during the online stage, we will resort to a weak imposition of (10.9) by Lagrange multipliers. More precisely, we will enforce weakly the normal component of (10.9) (i.e. $\boldsymbol{u}_h^{k+1}\cdot\boldsymbol{n} = D_t\eta_h^k$), while the tangential component of (10.9) (i.e. $\boldsymbol{u}_h^{k+1}\cdot\boldsymbol{\tau} = 0$) is already imposed strongly, since it is homogeneous and all basis functions in V_N satisfy it. Thus, during the offline stage we need to build an additional snapshot matrix of the Lagrange multipliers, in order to carry out a POD to obtain a reduced Lagrange multipliers space. The traction on the interface, which can be evaluated as the residual of (10.8) for test functions $\boldsymbol{v}_h := v_h\boldsymbol{n}$ which do not vanish on the interface, is indeed the Lagrange multiplier to (10.9). Therefore, we build an additional snapshot matrix

$$S_\lambda = [\underline{\boldsymbol{\lambda}}_h^1|\dots|\underline{\boldsymbol{\lambda}}_h^K] \in \mathbb{R}^{N_h^{\boldsymbol{u}}\times K},$$

containing the FE degrees of freedom corresponding to the residual of (10.8) for test functions $\boldsymbol{v}_h := v_h\boldsymbol{n}$ that do not vanish on the interface, compute a POD, and, similarly to the previous section, obtain a reduced space $L_N^{(1)}$ as the space spanned by the first N_λ left singular vectors. During the POD computation by the method of snapshots we employ the L^2 inner product on the interface as weight. We remark again here that the Lagrange multiplier approach is *not* actually used during the high-fidelity solution of the FSI system (in favor of a strong imposition), but rather the snapshot matrix S_λ is obtained as a post-processing of the obtained solution. In contrast, in the online stage the Lagrange multiplier approach will actually be used while solving the linear system associated to the reduced fluid viscous step, in order to impose weakly the coupling condition (10.9).

10.4.1.3 Online Stage

A reduced-order approximation of the FSI problem is then obtained by means of a Galerkin projection over the reduced spaces $V_N^{(1)}$, $Q_N^{(1)}$ and $E_N^{(1)}$, respectively, treating the coupling condition (10.9) with Lagrange multipliers in the reduced space $L_N^{(1)}$. Thus, the corresponding online stage of the POD–Galerkin method reads: for any $k = 1, \dots, K$

$1_N^{(1)}$. Explicit step (fluid viscous part), with weak imposition of coupling conditions through Lagrange multipliers: find $(\boldsymbol{u}_N^{k+1}, \lambda_N^{k+1}) \in V_N^{(1)} \times L_N^{(1)}$ such that:

$$\begin{cases} \int_\Omega \frac{\rho_f}{\Delta t} \boldsymbol{u}_N^{k+1} \cdot \boldsymbol{v}_N \, d\boldsymbol{x} & + \int_\Omega 2\mu_f \boldsymbol{\varepsilon}(\boldsymbol{u}_N^{k+1}) : \nabla \boldsymbol{v}_N \, d\boldsymbol{x} \\ & + \int_\Sigma \lambda_N^{k+1} \boldsymbol{n} \cdot \boldsymbol{v}_N \, ds = \int_\Omega \frac{\rho_f}{\Delta t} \boldsymbol{u}_N^k \cdot \boldsymbol{v}_N \, d\boldsymbol{x} \\ & - \int_\Omega \nabla p_N^k \cdot \boldsymbol{v}_N \, d\boldsymbol{x}, \\ \int_\Sigma \boldsymbol{u}_N^{k+1} \cdot \Upsilon_N \boldsymbol{n} \, ds & = \int_\Sigma D_t \eta_h^k \; \Upsilon_N \, ds, \end{cases}$$

for all $(\boldsymbol{v}_N, \Upsilon_N) \in V_N^{(1)} \times L_N^{(1)}$. We note that the boundary condition $\boldsymbol{u}_N^{k+1} \cdot \boldsymbol{n} = 0$ on Γ_{sym} is implicitly verified, since it is satisfied by any element in V_N.

$2_N^{(1)}$. Implicit step: for any $j = 0, \dots,$ until convergence:

$2.1_N^{(1)}$. Fluid projection substep, with Robin boundary conditions: find $p_N^{k+1,j+1} \in Q_N^{(1)}$ such that:

$$\int_\Omega \nabla p_N^{k+1,j+1} \cdot \nabla q_N \, d\boldsymbol{x} + \int_\Sigma \alpha_{Rob} \, p_N^{k+1,j+1} \, q_N \, ds = \\ - \int_\Omega \frac{\rho_f}{\Delta t} \operatorname{div} \boldsymbol{u}_N^{k+1} \, q_N \, d\boldsymbol{x} - \int_\Sigma \rho_f D_{tt} \eta_N^{k+1,j} \, q_N \, ds \\ + \int_\Sigma \alpha_{Rob} \, p_N^{k+1,j} \, q_N \, ds$$

for all $q_N \in Q_N^{(1)}$. The imposition of the boundary conditions $p_N^{k+1,j+1} = p_{in}(t)$ on Γ_{in} and $p_N^{k+1,j+1} = p_{out}(t)$ on Γ_{out}, although not automatically prescribed by the reduced space $Q_N^{(1)}$, can be easily treated by a lifting in 2.1_h without the need to introduce an additional Lagrange multiplier for the pressure, since the values to be imposed value do not depend on any reduced space, rather are known functions. The details are omitted for the sake of brevity; the interested reader is referred to [5] for more details.

$2.2_N^{(1)}$. Structure substep: find $\eta_N^{k+1,j+1} \in E_N^{(1)}$ such that:

$$\begin{aligned}\int_\Sigma \frac{\rho_s h_s}{\Delta t^2}\,\eta_N^{k+1,j+1}\,\zeta_N\,ds + \int_\Sigma c_1\,\partial_x \eta_N^{k+1,j+1}\,\partial_x\zeta_N\,ds \\ + \int_\Sigma c_0 \eta_N^{k+1,j+1}\,\zeta_N\,ds = \int_\Sigma \frac{\rho_s h_s}{\Delta t^2}\eta_N^k\,\zeta_N\,ds \\ + \int_\Sigma \frac{\rho_s h_s}{\Delta t} D_t\eta_N^k\,\zeta_N\,ds \\ - \int_\Sigma \boldsymbol{\sigma}(\boldsymbol{u}^{k+1}, p^{k+1,j+1})\boldsymbol{n}\cdot \zeta_N \boldsymbol{n}\,ds\end{aligned}$$

for all $\zeta_N \in E_N^{(1)}$. The boundary condition $\eta_N^{k+1,j+1} = 0$ on $\partial\Sigma$ is implicitly verified.

As for the high-fidelity model, a stopping criterion on the relative increment of the solution is employed to terminate step $2_N^{(1)}$.

Remark 2 (On Efficient Offline-Online Decoupling) The reduced-order problem $1_N^{(1)}$, $2.1_N^{(1)}$ and $2.2_N^{(1)}$ can easily account for an efficient offline-online decoupling, thanks to the linearity assumption in the problem formulation. For instance, the fluid mass term $\int_\Omega \frac{\rho_f}{\Delta t}\boldsymbol{u}_N^{k+1}\cdot \boldsymbol{v}_N\,d\boldsymbol{x}$ in $1_N^{(1)}$, is efficiently assembled at the end of the offline stage as

$$M_N^{(1)} := (Z_N^{\boldsymbol{u}})^T M_h Z_N^{\boldsymbol{u}},$$

and loaded during the online stage. Here $Z_N^{\boldsymbol{u}}$ is the matrix which contains the velocity basis functions $\{\boldsymbol{\varphi}_i\}_{i=1}^{N^{\boldsymbol{u}}}$ as columns, and M_h is the FE matrix corresponding to fluid mass term $\int_\Omega \frac{\rho_f}{\Delta t}\boldsymbol{u}_h^{k+1}\cdot \boldsymbol{v}_h\,d\boldsymbol{x}$ in 1_h. One can carry out a similar computational procedure for all terms in the reduced formulation $1_N^{(1)}$, $2.1_N^{(1)}$ and $2.2_N^{(1)}$.

In a more general (nonlinear, geometrical parametrized) setting one can resort to the empirical interpolation method [8] to recover an efficient offline-online splitting, as recently shown for FSI problems in [6].

Remark 3 (On Supremizer Enrichment: The Role of Pressure) In contrast to what is usually done in the reduced basis approximation of parametrized fluid dynamics problem (see [24, 25, 27], and also [5] for an extension to POD–Galerkin methods) and previous works on FSI [6, 9, 18], we do not employ in this case a supremizer enrichment of the velocity space to enforce inf-sup stability of the mixed velocity-pressure formulation. This is heuristically motivated by the fact that, even at the high-fidelity level, the Chorin-Temam scheme, in its pressure Poisson version, can be successfully applied to FE spaces that do not fulfill a (V, Q)-inf-sup condition [15], even though it may result in non-optimal error estimates.

Remark 4 (On Supremizer Enrichment: The Role of Lagrange Multiplier) Problem $1_N^{(1)}$ still features a saddle point structure. We remark that this structure is not due to the original problem, but rather due to our choice of coupling conditions by Lagrange multipliers. A drawback of ROM 1 is now apparent for what concerns the size of the reduced system $1_N^{(1)}$, which needs to be increased to $N_{\boldsymbol{u}} + N_{\boldsymbol{\lambda}}$ due to weak imposition of coupling condition. The ROM proposed in the next section has been devised to overcome this limitation, and results in a reduced explicit step of size $N_{\boldsymbol{u}}$. Moreover, a further increase in dimension would be required if we were willing to enrich the velocity space $V_N^{(1)}$ with supremizers corresponding to the inf-sup condition associated to problem $1_N^{(1)}$, i.e. solutions to

$$\int_{\Omega} \nabla \boldsymbol{s}^k \cdot \nabla \boldsymbol{v} = \int_{\Sigma} \boldsymbol{\lambda}_h^k \cdot \boldsymbol{v} \quad \forall \boldsymbol{v} \in V,$$

for all $k = 1, \dots, K$. In this work we do not carry out such enrichment since it would further increase the size of the reduced explicit step; nevertheless, a detailed investigation of the stability of $1_N^{(1)}$ with and without enrichment by $\boldsymbol{s}^k$ is an ongoing task and will be presented in a forthcoming work.

10.4.2 FSI ROM 2 Approach

As we have seen in the previous section, it is challenging to enforce (10.9) at the reduced-order level. The second reduced-order model proposed in this paper overcomes these difficulties performing a change of variable for the fluid velocity, namely defining an auxiliary unknown $\boldsymbol{z}^{k+1} : \Omega \to \mathbb{R}^2$ as

$$\boldsymbol{z}^{k+1} = \boldsymbol{u}^{k+1} - D_t \widehat{\eta}^k \boldsymbol{n}, \tag{10.11}$$

where $\widehat{\eta}^k$ is the solution of the following *harmonic extension* problem

$$-\Delta \widehat{\eta}^k = 0 \quad \text{in } \Omega,$$

subject to the following inhomogeneous boundary condition on the interface

$$\widehat{\eta}^k = \eta^k \quad \text{on } \Sigma,$$

and homogeneous boundary condition on the remaining boundaries. In this way, the coupling condition (10.9) is equivalent to

$$z^{k+1} = \mathbf{0} \quad \text{on } \Sigma,$$

for which no weak imposition by Lagrange multipliers is required.

10.4.2.1 Offline Stage

During the offline stage, the solution of the high-fidelity problem 1_h, 2.1_h and 2.2_h is first sought. Auxiliary unknowns z^{k+1} are then computed thanks to (10.11), for all $k = 0, \dots, K-1$. We then consider the following snapshot matrix

$$S_z = [\underline{\mathbf{z}}_h^1 | \dots | \underline{\mathbf{z}}_h^K] \in \mathbb{R}^{N_h^{\boldsymbol{u}} \times K},$$

and, similarly to Sect. 10.4.1.1, retain the first N_z POD modes in the reduced space $V_N^{(2)}$. Reduced spaces for fluid pressure and structure displacement are defined as in Sect. 10.4.1.1, $Q_N^{(2)} := Q_N^{(1)}$ and $E_N^{(2)} := E_N^{(1)} = \text{span}(\{\phi_l\}_{l=1}^{N^\eta})$. Moreover, harmonically extend each $\{\phi_l\}_{l=1}^{N^\eta}$ to $\{\widehat{\phi}_l\}_{l=1}^{N^\eta}$.

10.4.2.2 Online Stage

Similarly to Sect. 10.4.1.3, a reduced-order approximation of the FSI problem is now obtained by means of a Galerkin projection over the reduced spaces $V_N^{(2)}$, $Q_N^{(2)}$ and $E_N^{(2)}$, respectively, that is: for any $k = 1, \dots, K$

$1_N^{(2)}$. Explicit step (fluid viscous part), with change of variable for the fluid velocity: find $z_N^{k+1} \in V_N^{(2)}$ such that:

$$\int_\Omega \frac{\rho_f}{\Delta t} z_N^{k+1} \cdot \boldsymbol{v}_N \, d\boldsymbol{x} + \int_\Omega 2\mu_f \boldsymbol{\varepsilon}(z_N^{k+1}) : \nabla \boldsymbol{v}_N \, d\boldsymbol{x} =$$
$$\int_\Omega \frac{\rho_f}{\Delta t} \boldsymbol{u}_N^k \cdot \boldsymbol{v}_N \, d\boldsymbol{x} - \int_\Omega \nabla p_N^k \cdot \boldsymbol{v}_N \, d\boldsymbol{x},$$
$$- \int_\Omega \frac{\rho_f}{\Delta t} D_t \widehat{\eta}^{k_N} \, \boldsymbol{n} \cdot \boldsymbol{v}_N \, d\boldsymbol{x} - \int_\Omega 2\mu_f \boldsymbol{\varepsilon}(D_t \widehat{\eta}^{k_N} \, \boldsymbol{n}) : \nabla \boldsymbol{v}_N \, d\boldsymbol{x}$$

for all $\boldsymbol{v}_N \in V_N^{(2)}$. We note that boundary and *interface* conditions on z_N^{k+1} are implicitly verified, since they are homogeneous and satisfied by any element in $V_N^{(2)}$. Finally, for the sake of the implicit step, we define $\boldsymbol{u}_N^{k+1}$ as $z_N^{k+1} + D_t \widehat{\eta}^{k_N} \, \boldsymbol{n}$.

$2_N^{(2)}$. Implicit step: for any $j = 0, \ldots,$ until convergence:

$2.1_N^{(2)}$. Fluid projection substep, with Robin boundary conditions: as in $2.1_N^{(1)}$.

$2.2_N^{(2)}$. Structure substep: as in $2.1_N^{(2)}$. Finally, at convergence, harmonically extend η_N^{k+1} to $\widehat{\eta}_N^{k+1}$. Note that this can be easily carried out as the linear combination of the harmonically extended displacement basis $\{\widehat{\phi}_l\}_{l=1}^{N^\eta}$.

Remark 5 (On Efficient Offline-Online Decoupling) Similarly to Remark 2, an efficient offline-online decoupling can be obtained also in this case. Thanks to the definition of the harmonically extended displacements basis $\{\widehat{\phi}_l\}_{l=1}^{N^\eta}$ an efficient assembly of (e.g.) the right-hand side mass term $\int_\Omega \frac{\rho_f}{\Delta t} D_t \widehat{\eta}^{k_N} \boldsymbol{n} \cdot \boldsymbol{v}_N \, d\boldsymbol{x}$ can be obtained. We remark that this accounts for a negligible additional offline cost (solution of N^η harmonic extension problems) and no additional online cost, since the extension of η_N^{k+1} to $\widehat{\eta}_N^{k+1}$ does not actually require the solution of reduced problem, but rather a linear combination of $\{\widehat{\phi}_l\}_{l=1}^{N^\eta}$ once the coefficients of the structural unknown have been computed.

10.5 Numerical Comparison

In this section we summarize the numerical results obtained by the proposed reduced-order models. The values of constitutive and geometrical parameters are summarized in Table 10.1 [14]. The domain has been discretized with a 120×10 structured mesh, while the time-step is $\Delta t = 10^{-4}$ s. The final time is $T = 0.13$ s, so that $K = 1300$. T has been chosen to simulate the pressure wave propagation, just before wave reflection occurs. Numerical simulations are carried out using *RBniCS* [7, 17], an open-source reduced order modelling library developed at SISSA mathLab, built on top of *FEniCS* [20].

FSI ROM 1 Figure 10.1 shows the POD singular values and retained energy as a function of the number N of POD modes for FSI ROM 1. It can be noticed that the decay of the singular values of fluid velocity and structure displacement is slower than the one of fluid pressure; moreover, the first POD mode of fluid pressure retains a larger energy ($\approx 36\%$) than the first modes of structure displacement ($\approx 15\%$) and fluid velocity ($\approx 12\%$). Accordingly, the (relative) error analysis (Fig. 10.2a) shows

Table 10.1 Constitutive parameters for test case (from [14])

ρ_f	1 g/cm^3	μ_f	0.035 Poise	ρ_s	1.1 g/cm^3
E_s	0.75×10^6 dyn/cm^2	ν_s	0.5	c_1	$\frac{h_s E_s}{h_f^2\,(1-\nu_s^2)}$
c_0	$\frac{h_s E_s}{2\,(1+\nu_s)}$	L	6 cm	h_f	0.5 cm
h_s	0.1 cm	p_{in}	$10^4(1-\cos(\frac{2\pi t}{0.005}))\mathbf{1}_{t<0.005}$ dyn/cm^2	p_{out}	0 dyn/cm^2

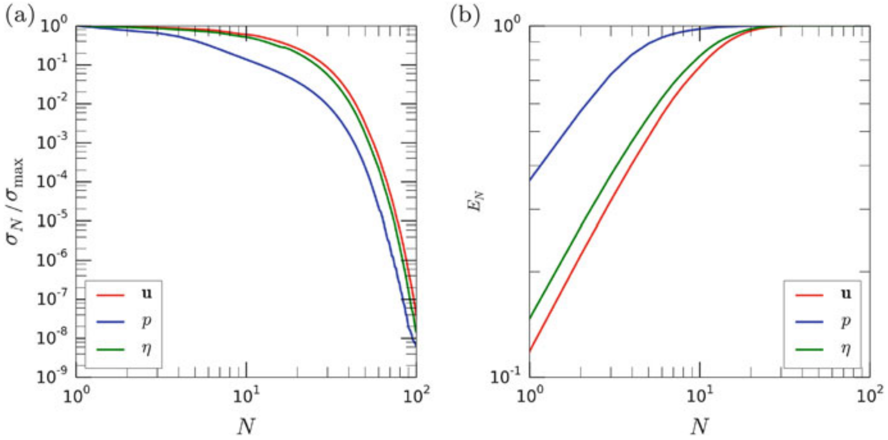


Fig. 10.1 Results of the offline stage of FSI ROM 1: (**a**) POD singular values and (**b**) retained energy as a function of the number N of POD modes for fluid velocity, fluid pressure, and solid displacement

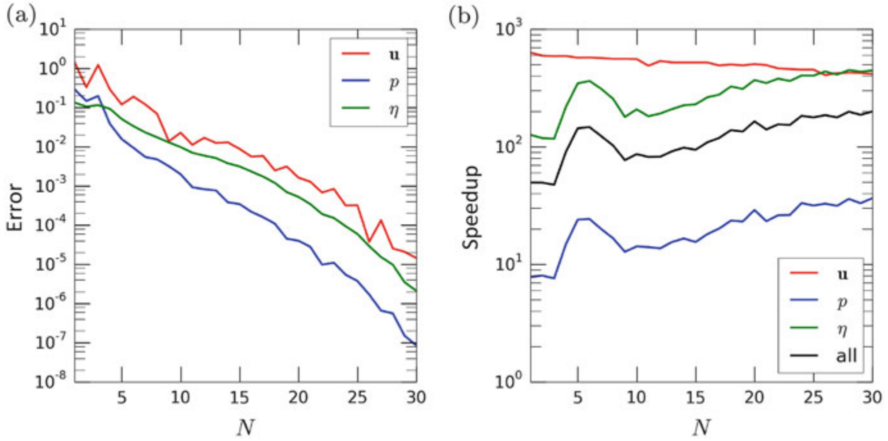


Fig. 10.2 (**a**) Error analysis and (**b**) speedup analysis of FSI ROM 1, as a function of the number N of POD modes for fluid velocity, fluid pressure, and solid displacement

that the reduced solution converges to the high-fidelity one, and that (except for small N), the relative error on the pressure is smaller than the displacement one, which is smaller than the velocity relative error. Employing only $N = 30$ POD modes out of the $K = 1300$ snapshots, velocity, displacement and pressure relative errors are of the order of 10^{-4}, 10^{-5} and 10^{-7}, respectively. Figure 10.2b shows that the overall speedup for large values of N is of at least two order of magnitudes. In particular, speedup for the explicit step is approximately constant for increasing N, while the speedup for the implicit steps is increasing for larger N since the number

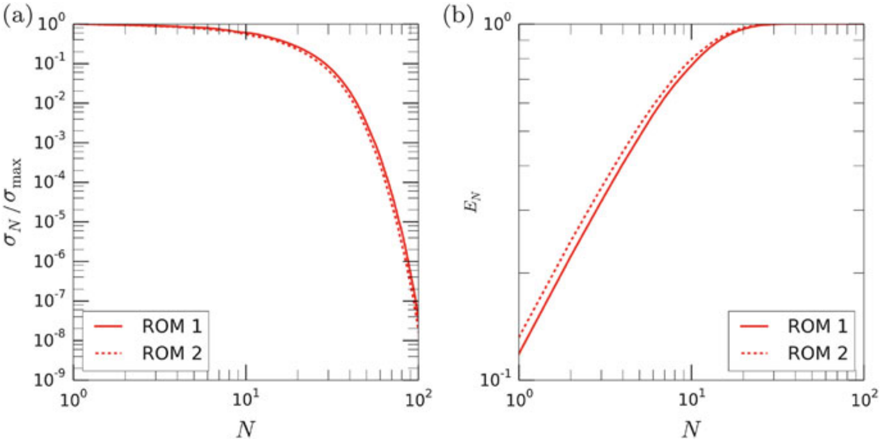


Fig. 10.3 Comparison of the offline stage of FSI ROMs 1 and 2: (**a**) POD singular values and (**b**) retained energy as a function of the number N of POD modes for fluid velocity $\boldsymbol{u}$ (FSI ROM 1) and auxiliary fluid velocity z (FSI ROM 2)

of required iterations for the implicit step decreases with N (see Fig. 10.7b and c for the maximum and average number of iterations, respectively).

FSI ROM 2 Figure 10.3 shows a comparison of the offline stage for ROMs 1 and 2. From Fig. 10.3b it can be noticed that, after the change of variable, the first auxiliary velocity z POD mode of FSI ROM 2 retains approximately 1.5% more energy than the first velocity $\boldsymbol{u}$ POD mode. The remaining variables are omitted since the pressure and displacement bases are the same as in FSI ROM 1. Moreover, Fig. 10.4a shows that the left-hand side matrix of the fluid viscous step $1_N^{(2)}$ (FSI ROM 2) is characterized by a condition number which is, for all N, at least 10 order of magnitude smaller than the one of $1_N^{(1)}$ (FSI ROM 1). The combination of these two remarks justifies the improvement in the error analysis for the velocity variables, shown in Fig. 10.5b. On average, the relative error on the velocity unknown obtained by FSI ROM 1 is seven times the one obtained by FSI ROM 2. Relative errors for the remaining unknowns are omitted because they are comparable among FSI ROMs 1 and 2. Moreover, we show in Fig. 10.6 the error analysis for the interface stress. The plot clearly shows that FSI ROM 2 provides a better approximation of the interface stress for $N > 20$. Online performance (Fig. 10.7) are comparable to the ones obtained by FSI ROM 1. This is due to the fact that (i) the number of iterations to reach convergence in step $2_N^{(1)}$ and $2_N^{(2)}$ are the same as maximum values (Fig. 10.4b) and comparable on average (Fig. 10.4c), and (ii) the time to solve the explicit step does not depend strongly on N (Fig. 10.7b), even though step $1_N^{(2)}$ (FSI ROM 2) requires the solution of a linear system of size $N \times N$ (rather than $2N \times 2N$ for step $1_N^{(1)}$ (FSI ROM 1)).

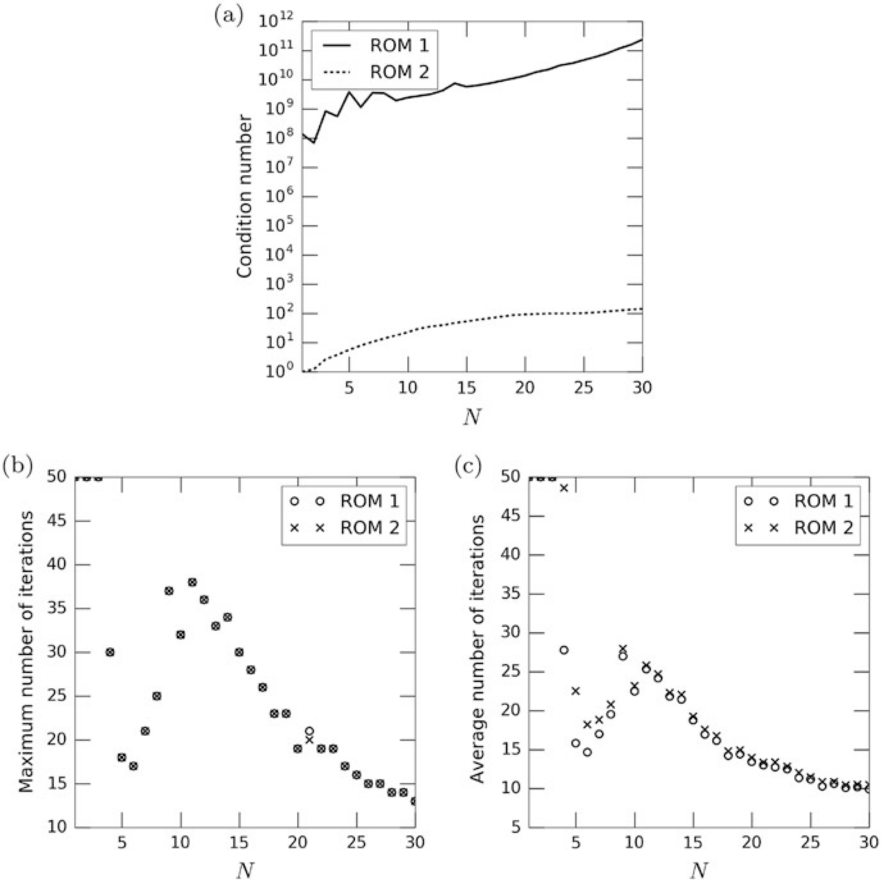


Fig. 10.4 Comparison of the condition number of the left-hand side matrix of $1_N^{(1)}$ and $1_N^{(2)}$ ((**a**)) and of the maximum ((**b**)) and average ((**c**)) number of iterations required by $2_N^{(1)}$ and $2_N^{(2)}$, as a function of the number N of POD modes for fluid velocity, fluid pressure, and solid displacement

10.6 Conclusions and Perspectives

Two semi-implicit reduced-order models for FSI problems have been proposed in this work, based on a POD–Galerkin approximation of an operator splitting semi-implicit high-fidelity scheme. FSI ROM 1 is a standard Galerkin projection over the reduced spaces generated by POD. No supremizer enrichment is required, thanks to the operator splitting approach. Even though FSI ROM 1 shows good performance in terms of error analysis, its major drawbacks (when compared to FSI ROM 2) are related to the weak imposition of (10.9) by Lagrange multipliers. Numerical results of the previous section have shown that this is detrimental for several

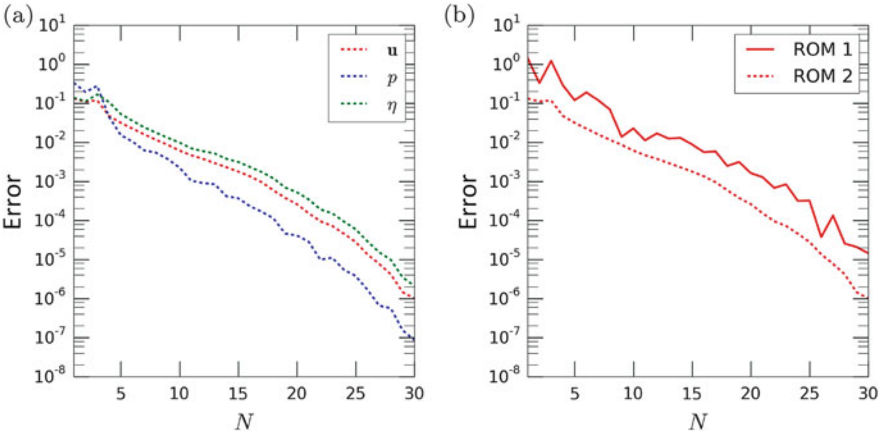


Fig. 10.5 Error analysis of FSI ROM 2, as a function of the number N of POD modes for fluid velocity, fluid pressure, and solid displacement. (**a**) Error analysis of FSI ROM 2. (**b**) Comparison of the error for fluid velocity unknown of FSI ROMs 1 and 2

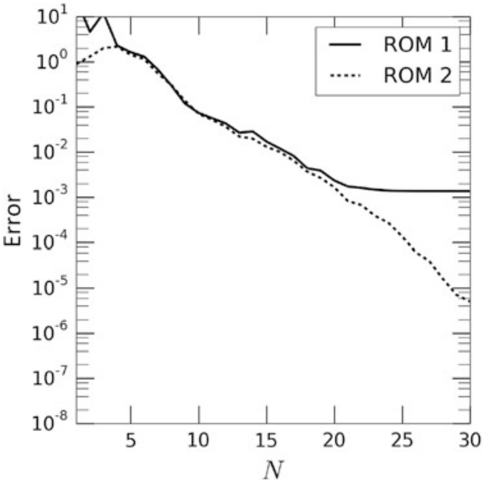


Fig. 10.6 Error analysis of the interface stress for FSI ROMs 1 and 2, as a function of the number N of POD modes for fluid velocity, fluid pressure, and solid displacement

aspects of the ROM: increased system dimension of the fluid explicit step, increased condition number of the fluid explicit step, increased error for the velocity. FSI ROM 2, instead, stems from the idea that (10.9) can be easily imposed in a reduced-order framework by performing the change of variable (10.11). In this way, all the detrimental effects of FSI ROM 1 are remedied. Moreover, better properties in terms of POD retained energy are also obtained. The combination of these two factors results in a better approximation of the fluid velocity. In particular, in FSI ROM 2,

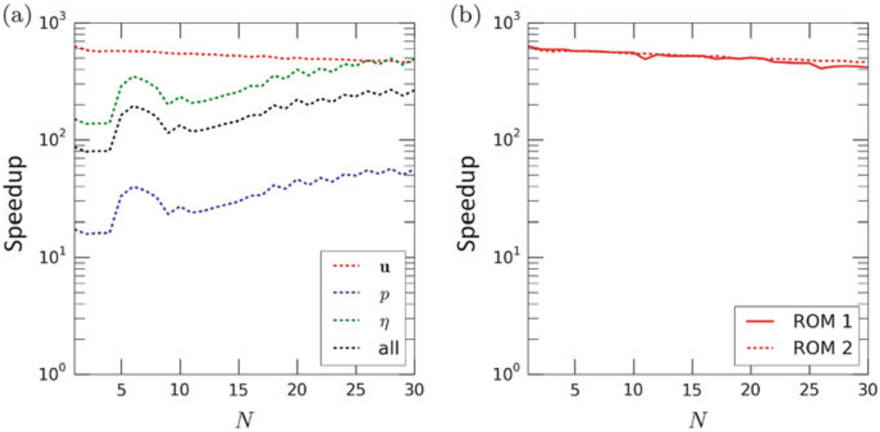


Fig. 10.7 Speedup analysis of FSI ROM 2, as a function of the number N of POD modes for fluid velocity, fluid pressure, and solid displacement. (**a**) Speedup analysis of FSI ROM 2. (**b**) Comparison of the speedup for fluid velocity unknown of FSI ROMs 1 and 2

we try to separate in the fluid velocity the fluid-structure interaction component from the pure fluid part. Even though FSI ROM 1 suffers several drawbacks when compared to FSI ROM 2 (especially for what concerns the increased condition number), the approach proposed by FSI ROM 1 can be more easily integrated with existing reduced order modelling capabilities for fluid problems, since it does not require to change the existing computations of fluid velocity basis functions. Nevertheless, especially for FSI ROM 1, a more detailed analysis of enrichment procedures shall be carried out to further investigate the stability of the resulting reduced problem, due to two saddle point structures to be taken into account (see Remarks 3 and 4). Future work will concern ROMs that are better able to face the hyperbolic nature of the problem. A more efficient separation at the reduced-order level of the parabolic and hyperbolic components of the system may decrease the number of basis functions required to obtain an accurate reduced description of the FSI problem.

Acknowledgements We acknowledge the support by European Union Funding for Research and Innovation—Horizon 2020 Program—in the framework of European Research Council Executive Agency: H2020 ERC CoG 2015 AROMA-CFD project 681447 “Advanced Reduced Order Methods with Applications in Computational Fluid Dynamics” and the PRIN project “Mathematical and numerical modelling of the cardiovascular system, and their clinical applications”. We also acknowledge the INDAM-GNCS project “Tecniche di riduzione della complessità computazionale per le scienze applicate” and INDAM-GNCS young researchers project “Numerical methods for model order reduction of PDEs”.

References

1. Amsallem, D., Cortial, J., Farhat, C.: Towards real-time computational-fluid-dynamics-based aeroelastic computations using a database of reduced-order information. AIAA J. **48**(9), 2029–2037 (2010)
2. Astorino, M., Chouly, F., Fernández, M.A.: Robin based semi-implicit coupling in fluid-structure interaction: Stability analysis and numerics. SIAM J. Sci. Comput. **31**(6), 4041–4065 (2010)
3. Badia, S., Nobile, F., Vergara, C.: Fluid–structure partitioned procedures based on Robin transmission conditions. J. Comput. Phys. **227**(14), 7027–7051 (2008)
4. Badia, S., Quaini, A., Quarteroni, A.: Splitting methods based on algebraic factorization for fluid-structure interaction. SIAM J. Sci. Comput. **30**(4), 1778–1805 (2008)
5. Ballarin, F., Manzoni, A., Quarteroni, A., Rozza, G.: Supremizer stabilization of POD–Galerkin approximation of parametrized steady incompressible Navier–Stokes equations. Int. J. Numer. Methods Eng. **102**(5), 1136–1161 (2015)
6. Ballarin, F., Rozza, G.: POD–Galerkin monolithic reduced order models for parametrized fluid-structure interaction problems. Int. J. Numer. Methods Fluids **82**(12), 1010–1034 (2016)
7. Ballarin, F., Sartori, A., Rozza, G.: RBniCS – reduced order modelling in fenics. http://mathlab.sissa.it/rbnics (2016)
8. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. C.R. Math. **339**(9), 667–672 (2004)
9. Colciago, C.M.: Reduced order fluid-structure interaction models for haemodynamics applications. Ph.D. Thesis, École Polytechnique Fédérale de Lausanne, N. 6285 (2014)
10. Fernández, M.A.: Incremental displacement-correction schemes for incompressible fluid-structure interaction. Numer. Math. **123**(1), 21–65 (2013)
11. Fernández, M.A., Gerbeau, J.F., Grandmont, C.: A projection semi-implicit scheme for the coupling of an elastic structure with an incompressible fluid. Int. J. Numer. Methods Eng. **69**(4), 794–821 (2007)
12. Fernández, M.A., Landajuela, M., Mullaert, J., Vidrascu, M.: Robin-Neumann schemes for incompressible fluid-structure interaction. Domain Decomposition Methods in Science and Engineering, vol. XXII. pp. 65–76. Springer, Cham (2016)
13. Fernández, M.A., Mullaert, J., Vidrascu, M.: Explicit Robin–Neumann schemes for the coupling of incompressible fluids with thin-walled structures. Comput. Methods Appl. Mech. Eng. **267**, 566–593 (2013)
14. Formaggia, L., Gerbeau, J., Nobile, F., Quarteroni, A.: On the coupling of 3D and 1D Navier–Stokes equations for flow problems in compliant vessels. Comput. Methods Appl. Mech. Eng. **191**(6–7), 561–582 (2001)
15. Guermond, J.L., Quartapelle, L.: On stability and convergence of projection methods based on pressure poisson equation. Int. J. Numer. Methods Fluids **26**(9), 1039–1053 (1998)
16. Guidoboni, G., Glowinski, R., Cavallini, N., Canic, S.: Stable loosely-coupled-type algorithm for fluid–structure interaction in blood flow. J. Comput. Phys. **228**(18), 6916–6937 (2009)
17. Hesthaven, J.S., Rozza, G., Stamm, B.: Certified reduced basis methods for parametrized partial differential equations. SpringerBriefs in Mathematics. Springer, New York (2015)
18. Lassila, T., Manzoni, A., Quarteroni, A., Rozza, G.: A reduced computational and geometrical framework for inverse problems in hemodynamics. Int. J. Numer. Methods Biomed. Eng. **29**(7), 741–776 (2013)
19. Lassila, T., Quarteroni, A., Rozza, G.: A reduced basis model with parametric coupling for fluid-structure interaction problems. SIAM J. Sci. Comput. **34**(2), A1187–A1213 (2012)
20. Logg, A., Mardal, K.A., Wells, G.N.: Automated Solution of Differential Equations by the Finite Element Method. Springer, Berlin (2012)

21. Quarteroni, A., Formaggia, L.: Mathematical modelling and numerical simulation of the cardiovascular system. In: Computational Models for the Human Body. Handbook of Numerical Analysis, vol. 12, pp. 3–127. Elsevier, Amsterdam (2004)
22. Quarteroni, A., Tuveri, M., Veneziani, A.: Computational vascular fluid dynamics: problems, models and methods. Comput. Vis. Sci. **2**(4), 163–197 (2000). doi:10.1007/s007910050039. http://dx.doi.org/10.1007/s007910050039
23. Quarteroni, A., Valli, A.: Numerical Approximation of Partial Differential Equations, vol. 23. Springer, Berlin (2008)
24. Rozza, G.: Reduced basis methods for Stokes equations in domains with non-affine parameter dependence. Comput. Vis. Sci. **12**(1), 23–35 (2009)
25. Rozza, G., Huynh, D.B.P., Manzoni, A.: Reduced basis approximation and a posteriori error estimation for stokes flows in parametrized geometries: roles of the inf-sup stability constants. Numer. Math. **125**(1), 115–152 (2013)
26. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Arch. Comput. Meth. Eng. **15**, 1–47 (2007)
27. Rozza, G., Veroy, K.: On the stability of the reduced basis method for Stokes equations in parametrized domains. Comput. Methods Appl. Mech. Eng. **196**(7), 1244–1260 (2007)

Chapter 11
True Error Control for the Localized Reduced Basis Method for Parabolic Problems

Mario Ohlberger, Stephan Rave, and Felix Schindler

Abstract We present an abstract framework for a posteriori error estimation for approximations of scalar parabolic evolution equations, based on elliptic reconstruction techniques (Makridakis and Nochetto, SIAM J. Numer. Anal. 41(4):1585–1594, 2003. doi:10.1137/S0036142902406314; Lakkis and Makridakis, Math. Comput. 75(256):1627–1658, 2006. doi:10.1090/S0025-5718-06-01858-8; Demlow et al., SIAM J. Numer. Anal. 47(3):2157–2176, 2009. doi:10.1137/070708792; Georgoulis et al., SIAM J. Numer. Anal. 49(2):427–458, 2011. doi:10.1137/080722461). In addition to its original application (to derive error estimates on the discretization error), we extend the scope of this framework to derive offline/online decomposable a posteriori estimates on the model reduction error in the context of Reduced Basis (RB) methods. In addition, we present offline/online decomposable a posteriori error estimates on the full approximation error (including discretization as well as model reduction error) in the context of the localized RB method (Ohlberger and Schindler, SIAM J. Sci. Comput. 37(6):A2865–A2895, 2015. doi:10.1137/151003660). Hence, this work generalizes the localized RB method with true error certification to parabolic problems. Numerical experiments are given to demonstrate the applicability of the approach.

11.1 Introduction

We are interested in efficient and certified numerical approximations of parabolic parametric problems, such as: given a Gelfand triple of suitable Hilbert spaces $Q \subset H \subset Q'$, an end time $T_{\text{end}} > 0$, initial data $p_0 \in Q$ and right hand side $f \in H$, for a

This work has been supported by the German Federal Ministry of Education and Research (BMBF) under contract number 05M13PMA.

M. Ohlberger • S. Rave • F. Schindler (✉)
Institute for Computational and Applied Mathematics, University of Münster, Einsteinstrasse 62, 48149, Münster, Germany
e-mail: mario.ohlberger@uni-muenster.de; stephan.rave@uni-muenster.de; felix.schindler@uni-muenster.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_11

parameter $\mu \in \mathscr{P}$ find $p(\cdot;\mu) \in L^2(0,T_{\text{end}};Q)$ with $\partial_t p(\cdot;\mu) \in L^2(0,T_{\text{end}};Q')$, such that $p(0;\mu) = p_0$ and

$$\langle \partial_t p(t;\mu), q\rangle + b\big(p(t;\mu), q;\mu\big) = (f,q)_H \qquad \text{for all } q \in Q, \tag{11.1}$$

where $\mathscr{P} \subset \mathbb{R}^\rho$ for $\rho \in \mathbb{N}$ denotes the set of admissible parameters and b denotes a parametric elliptic bilinear form (see Sect. 11.2 for details).

We consider grid-based approximations $p_h(\mu) \in Q_h$ of $p(\mu) \in Q$, obtained by formulating (11.1) in terms of a discrete approximation space $Q_h \subset H$ (think of Finite Elements or Finite Volumes) where b is replaced by a discrete counterpart acting on Q_h (e.g. in case of nonconforming approximations).

Efficiency of such an approximation for a single parameter is usually associated with minimal computational effort, obtained by adaptive grid refinement using localizable and reliable error estimates (see [17] and the references therein). For parametric problems, however, where one is interested in approximating (11.1) for many parameters, efficiency is related to an overall computational cost that is minimal compared to the combined cost of separate approximations for each parameter. To this end one employs model reduction with reduced basis (RB) methods, where one usually considers a common approximation space Q_h for all parameters (with the notable exceptions [2, 18]) and where one iteratively builds a reduced approximation space $Q_{\text{red}} \subset Q_h$ by an adaptive greedy search, the purpose of which is to capture the manifold of solutions of (11.1): $\{p(t;\mu) \in Q \,|\, t \in [0,T_{\text{end}}], \mu \in \mathscr{P}\}$; we refer to the monographs [7, 15, 16] and the references therein. One obtains a reduced problem by Galerkin projection of all quantities onto Q_{red} and, given a suitable parametrization of the problem, the assembly of the reduced problem allows for an offline/online decomposition such that a reduced solution $p_{\text{red}}(\mu) \in Q_{\text{red}}$ for a parameter $\mu \in \mathscr{P}$ can be efficiently computed with a computational effort independent of the dimension of Q_h. To assess the quality of the reduced solution and to steer the greedy basis generation, RB methods traditionally rely on residual based a posterior error estimates on the *model reduction error* $e_{\text{red}}(\mu) := p_h(\mu) - p_{\text{red}}(\mu)$, $\|e_{\text{red}}(\mu)\| \leq \eta_{\text{red}}(\mu)$, with the drawback that usually no information on the *discretization error* $e_h(\mu) := p(\mu) - p_h(\mu)$ is available during the online phase of the computation.

In contrast, we are interested in approximations of (11.1) which are efficient in the parametric sense as well as certified in the sense that we have access to an efficiently computable estimate on the *full approximation error* $e_{h,\text{red}}(\mu) := p(\mu) - p_{\text{red}}(\mu)$, including the discretization as well as the model reduction error: $\|e_{h,\text{red}}(\mu)\| \leq \eta_{h,\text{red}}(\mu)$.

For elliptic problems such an estimate is available for the localized RB multiscale method (LRBMS) [1, 14], the idea of which is to couple spatially localized reduced bases associated with subdomains of the physical domain. In addition to computational benefits due to the localization, the LRBMS also allows to adaptively enrich the local reduced bases online by solving local corrector problems, given a localizable error estimate. Apart from the LRBMS [13], we are only aware of [2] and [18], where the full approximation error is taken into account in the context of RB methods.

In an instationary setting, localized RB methods were first applied in the context of two-phase flow in porous media [8] and to parabolic problems such as (11.1) in the context of Lithium-Ion Battery simulations [12], yet in either case without error control. In contrast, here we present the fully certified localized RB method for parabolic problems by equipping it with suitable a posteriori error estimates. As argued above, it is beneficial to have access to several error estimates which can be evaluated efficiently during the online phase: for instance to later enable online adaptive basis enrichment, one could (i) solve local corrector problems, given η_{red}, whenever the reduced space is not rich enough; or one could (ii) locally adapt the grid, given $\eta_{h,\text{red}}$, whenever Q_h is insufficient.

Therefore, we present a general framework for a posteriori error estimation for parabolic problems, which will enable us to obtain either of the above estimates. It is based on the elliptic reconstruction technique, introduced for several discretizations and norms in [3, 5, 9, 10]. In this contribution we reformulate this approach in an abstract setting, allowing for a novel application in the context of RB methods. In particular, this technique allows to reuse existing a posteriori error estimates for elliptic diffusion problems.

11.2 General Framework for A Posteriori Error Estimates

In the following presentation we mainly follow [5], reformulating it in an abstract Hilbert space setting and slightly extending it by allowing non-symmetric bilinear forms. We drop the parameter dependency in this section to simplify the notation.

Definition 1 (Abstract Parabolic Problem) Let Q be a Hilbert space, densely embedded in another Hilbert space H (possibly $Q = H$), and let $\widetilde{Q} \subseteq H$ be a finite dimensional approximation space for Q, not necessarily contained in Q. Denote by $(\cdot,\cdot)$, $\|\cdot\|$ the H-inner product and the norm induced by it.

Let $f \in H$, and let $b : (Q + \widetilde{Q}) \times (Q + \widetilde{Q}) \to \mathbb{R}$ be a bilinear form which is continuous and coercive on Q. Let further $|||\cdot|||$ be a norm over $Q + \widetilde{Q}$, which coincides with the square root of the symmetric part of b over Q.

Our goal is to bound the error $e(t) := p(t) - \tilde{p}(t)$ between the true (analytical) solution $p \in L^2(0, T_{\text{end}}; Q)$, $\partial_t p \in L^2(0, T_{\text{end}}; Q')$ of (11.1), where the duality pairing $\langle \partial_t p(t), q \rangle$ is induced by the H-scalar product via the Gelfand triple $Q \subseteq H = H' \subseteq Q'$, and the $\widetilde{Q}$-Galerkin approximation $\tilde{p} \in L^2(0, T_{\text{end}}, \widetilde{Q})$, $\partial_t \tilde{p} \in L^2(0, T_{\text{end}}, \widetilde{Q})$, solution of

$$(\partial_t \tilde{p}(t), \tilde{q}) + b(\tilde{p}(t), \tilde{q}) = (f, \tilde{q}) \qquad \text{for all } \tilde{q} \in \widetilde{Q}. \tag{11.2}$$

Definition 2 (Elliptic Reconstruction) Denote by $\widetilde{\Pi}$ the H-orthogonal projection onto $\widetilde{Q}$. For $\tilde{q} \in \widetilde{Q}$, define the elliptic reconstruction $\mathscr{E}(\tilde{q}) \in Q$ of $\tilde{q}$ to be the unique solution of the variational problem

$$b(\mathscr{E}(\tilde{q}), q') = (B(\tilde{q}) - \widetilde{\Pi}(f) + f, q') \qquad \text{for all } q' \in Q, \tag{11.3}$$

where $B(\tilde{q}) \in \widetilde{Q}$ is the H-inner product Riesz representative of the functional $b(\tilde{q}, \cdot)$, i.e., $(B(\tilde{q}), \tilde{q}') = b(\tilde{q}, \tilde{q}')$ for all $\tilde{q}' \in \widetilde{Q}$. Note that $\mathscr{E}(\tilde{q})$ is well-defined, due to the coercivity of b on Q.

The following central property of the elliptic reconstruction follows immediately from its definition:

Proposition 1 *$\tilde{q}$ is the $\widetilde{Q}$-Galerkin approximation of the solution $\mathscr{E}(\tilde{q})$ of the weak problem (11.3) in the sense that $\tilde{q}$ satisfies*

$$b(\tilde{q}, \tilde{q}') = (\tilde{w} - \widetilde{\Pi}(f) + f, \tilde{q}') \qquad \textit{for all } \tilde{q}' \in \widetilde{Q}.$$

Assume that for each t we have a decomposition $\tilde{p}(t) =: \tilde{p}^c(t) + \tilde{p}^d(t)$ (not necessarily unique) where $\tilde{p}^c(t) \in Q$, $\tilde{p}^d(t) \in \widetilde{Q}$ are the conforming and non-conforming parts of $\tilde{p}(t)$. We consider the following error quantities:

$$\rho(t) := p(t) - \mathscr{E}(\tilde{p}(t)), \qquad \varepsilon(t) := \mathscr{E}(\tilde{p}(t)) - \tilde{p}(t),$$
$$e^c(t) := p(t) - \tilde{p}^c(t), \qquad \varepsilon^c(t) := \mathscr{E}(\tilde{p}(t)) - \tilde{p}^c(t).$$

Theorem 1 (Abstract Semi-Discrete Error Estimate) *Let $C := (2\,|||b|||^2 + 1)^{1/2}$, where $|||b|||$ denotes the continuity constant of b on Q w.r.t. $|||\cdot|||$, then*

$$\begin{aligned}\|e\|_{L^2(0,T_{end};|||\cdot|||)} \leq & \|e^c(0)\| + \sqrt{3}\|\partial_t \tilde{p}^d\|_{L^2(0,T_{end};|||\cdot|||_{Q,-1})} \\ & + (C+1)\cdot \|\varepsilon\|_{L^2(0,T_{end};|||\cdot|||)} + C \cdot \|\tilde{p}^d\|_{L^2(0,T_{end};|||\cdot|||)}.\end{aligned}$$

Proof (cf. [5]) For each $q \in Q$, we have the error identity

$$\langle \partial_t e(t), q\rangle + b(\rho(t), q) = 0, \tag{11.4}$$

using the definition of ρ, the properties of the elliptic reconstruction and the fact, that p solves (11.1). Testing with $e^c(t)$ and applying Young's inequality then yields

$$\partial_t \|e^c(t)\|^2 + |||\rho(t)|||^2 \leq 3\left|\left|\left|\partial_t \tilde{p}^d(t)\right|\right|\right|^2_{Q,-1} + (2\,|||b|||^2 + 1)\cdot |||\varepsilon^c(t)|||^2. \tag{11.5}$$

Hence, the claim follows by integrating (11.5) from 0 to T_{end} and using the triangle inequalities $|||e(t)||| \leq |||\rho(t)||| + |||\varepsilon(t)|||$ and $|||\varepsilon^c(t)||| \leq |||\varepsilon(t)||| + \left|\left|\left|\tilde{p}^d(t)\right|\right|\right|$. □

Remark 1 According to Proposition 1, the term $|||\varepsilon(t)|||$ can be bounded using any available a posteriori error estimator for the elliptic equation (11.3). The term $\left|\left|\left|\partial_t \tilde{p}^d(t)\right|\right|\right|_{Q,-1}$ can be bounded by $C^b_{H,Q}\|\partial_t \tilde{p}^d(t)\|$ using the Cauchy-Schwarz inequality, where $C^b_{H,Q}$ is a constant such that $\|q\| \leq C^b_{H,Q}\,|||q|||$ for all $q \in Q$.

It is straightforward to modify the estimate in Theorem 1 for semi-discrete solutions $\tilde{p}(t)$ to take the time discretization error into account:

Corollary 1 *Let $\tilde{p} \in L^2(0, T_{end}, \widetilde{Q})$, $\partial_t \tilde{p} \in L^2(0, T_{end}, \widetilde{Q})$ be an arbitrary discrete approximation of $p(t)$, not necessarily satisfying (11.2). Let $\mathscr{R}_T[\tilde{p}](t) \in \widetilde{Q}$ denote the $\widetilde{Q}$-Riesz representative w.r.t. the H-inner product of the time-stepping residual of $\tilde{p}(t)$, i.e.*

$$(\mathscr{R}_T[\tilde{p}](t), \tilde{q}) = (\partial_t \tilde{p}(t), \tilde{q}) + b(\tilde{p}(t), \tilde{q}) - (f, \tilde{q}) \qquad \forall \tilde{q} \in \widetilde{Q}.$$

Then, with $C := (3\, |||b|||^2 + 2)^{1/2}$, the following error estimate holds:

$$\begin{aligned} \|e\|_{L^2(0,T_{end};|||\cdot|||)} \leq \|e^c(0)\| &+ 2\|\partial_t \tilde{p}^d\|_{L^2(0,T_{end};|||\cdot|||_{Q,-1})} \\ &+ (C+1)\cdot \|\varepsilon\|_{L^2(0,T_{end};|||\cdot|||)} + C\cdot \|\tilde{p}^d\|_{L^2(0,T_{end};|||\cdot|||)} \\ &+ 2C^b_{H,Q}\cdot \|\mathscr{R}_T[\tilde{p}]\|_{L^2(0,T_{end};H)}. \end{aligned} \tag{11.6}$$

Proof Since (11.4) no longer holds, we gain $\mathscr{R}_T[\tilde{p}](t)$ as an additional source term in the error equation:

$$\langle \partial_t e(t), q\rangle + b(\rho(t), q) = (-\mathscr{R}_T[\tilde{p}](t), q).$$

The statement follows using the same line of argument as in the proof of Theorem 1, taking the additional term into account. □

Example 1 (Implicit Euler Time Stepping) Let $n_t \in \mathbb{N}$ be the number of time steps and $\Delta_t := T_{\text{end}}/n_t$ the (fixed) time step size. Let $\tilde{p}(t)$ be the $\widetilde{Q}$-valued piecewise linear function with supporting points $\tilde{p}(n\cdot\Delta_t) =: \tilde{p}^n$, $n = 0, \ldots n_t$, such that $\tilde{p}^0 := p(0)$ and $\tilde{p}^n$ is defined for $n > 0$ as the solution of

$$\left(\frac{\tilde{p}^n - \tilde{p}^{n-1}}{\Delta_t}, \tilde{q}\right) + b(\tilde{p}^n, \tilde{q}) = (f, \tilde{q}) \qquad \forall \tilde{q} \in \widetilde{Q}.$$

We then have for $(n-1)\cdot\Delta_t \leq t \leq n\cdot t$ the equality

$$\mathscr{R}_T[\tilde{p}](t) = \frac{n\cdot\Delta_t - t}{\Delta_t} B(\tilde{p}^n - \tilde{p}^{n-1}).$$

Thus,

$$\|\mathscr{R}_T[\tilde{p}]\|_{L^2(0,T_{\text{end}};H)} = \left\{\sum_{n=1}^{n_t} \frac{\Delta_t}{3} \|B(\tilde{p}^n - \tilde{p}^{n-1})\|^2\right\}^{1/2}.$$

Similarly, we obtain for the other quantities in (11.6) the bounds

$$\|\varepsilon\|_{L^2(0,T_{\mathrm{end}};|||\cdot|||)} \le 2\left\{\sum_{n=0}^{n_t} \frac{\Delta_t}{3}\,|||\varepsilon^n|||^2\right\}^{1/2},$$

$$\|\tilde{p}^d\|_{L^2(0,T_{\mathrm{end}};|||\cdot|||)} \le 2\left\{\sum_{n=1}^{n_t} \frac{\Delta_t}{3}\,\left|\left|\left|\tilde{p}^{d,n}\right|\right|\right|^2\right\}^{1/2},$$

$$\|\partial_t\tilde{p}^d\|_{L^2(0,T_{\mathrm{end}};|||\cdot|||_{Q,-1})} \le \left\{\sum_{n=1}^{n_t} \frac{1}{\Delta_t}\,\left|\left|\left|\tilde{p}^{d,n} - \tilde{p}^{d,n-1}\right|\right|\right|^2_{Q,-1}\right\}^{1/2},$$

where $\varepsilon^n := \varepsilon(n\cdot\Delta_t)$, $\tilde{p}^{d,n} := \tilde{p}^d(n\cdot\Delta_t)$, $0 \le n \le n_t$.

Example 2 (Reduced Basis Approximation) We can directly apply Corollary 1 to obtain a posteriori estimates for standard reduced basis schemes. In this case, $Q = H$ will be some discrete 'truth' space and $\widetilde{Q} \subseteq Q$ the reduced approximation space. The Q and H-norms might be, in case of a conforming approximation, the $H^1_0(\Omega)$ and $L^2(\Omega)$ norms for some domain Ω. (11.6) then reduces to

$$\|e\|_{L^2(0,T_{\mathrm{end}};|||\cdot|||)} \le \|e(0)\| + (C+1)\cdot\|\varepsilon\|_{L^2(0,T_{\mathrm{end}};|||\cdot|||)} + 2C^b_{H,Q}\cdot\|\mathscr{R}_T(\tilde{p})\|_{L^2(0,T_{\mathrm{end}};H)}.$$

The elliptic error $|||\varepsilon|||_{L^2(0,T_{\mathrm{end}};Q)}$ could be bounded using a standard residual-based error estimator for (11.3). For parametric problems with affine parameter dependency, all appearing terms are easily offline/online decomposed.

11.3 Localized Reduced Basis Methods

We now return to the definition of the localized RB method for parabolic problems as follows.

The Continuous Problem Let $\Omega \subset \mathbb{R}^d$ for $d = 1, 2, 3$ denote a bounded connected domain with polygonal boundary $\partial\Omega$ and, following the notation of Sect. 11.1, let $H = L^2(\Omega)$ and $Q = H^1_0(\Omega)$. We consider problem (11.1) with the parametric bilinear form b, defined over Q, as

$$b(p, q; \mu) = \int_\Omega (\lambda(\mu)\kappa_\varepsilon\nabla p)\cdot\nabla q \qquad \text{for}\, p, q \in H^1(\Omega), \mu \in \mathscr{P}, \tag{11.7}$$

given data functions $\kappa_\varepsilon \in [L^\infty(\Omega)]^{d\times d}$ and $\lambda : \mathscr{P} \to L^\infty(\Omega)$. For λ and κ_ε, such that $\lambda(\mu)\kappa_\varepsilon \in [L^\infty(\Omega)]^{d\times d}$ is bounded from below (away from 0) and above for all $\mu \in \mathscr{P}$, the bilinear form $b(\cdot,\cdot;\mu)$ is continuous and coercive with respect to Q for all $\mu \in \mathscr{P}$. Thus, a unique solution $p(\cdot;\mu) \in L^2(0, T_{\text{end}}; Q)$ of problem (11.1) exists for all $\mu \in \mathscr{P}$, if f is bounded.

We continue with the definition of the discretization in order to define the approximation space $\tilde{Q}$, to extend the definition of b onto $\tilde{Q}$ and to introduce the relevant norms.

The main idea of localized RB methods is to partition the physical domain Ω into subdomains in the spirit of domain decomposition methods and to generate a local reduced basis on each subdomain, as opposed to a single reduced basis with global support. Coupling across subdomains is achieved by a symmetric weighted interior penalty discontinuous Galerkin (SWIPDG) scheme [4] for the high-dimensional as well as the reduced discretization.

The Discretization To discretize (11.1) we require two nested partitions of Ω: a coarse one, $\mathscr{T}_H$ with elements (subdomains) $T \in \mathscr{T}_H$, and a fine one, τ_h with elements $t \in \tau_h$ (note that we use t to denote elements of the computational grids, not to be confused with the time t). Within each subdomain $T \in \mathscr{T}_H$ we allow for any local approximation of Q and b by discrete counterparts $Q_h^{k,T}$ and b_h^T of order $k \geq 1$, associated with the local grid $\tau_h^T := T \cap \tau_h \subset \tau_h$. In particular we consider (i) local conforming approximations by setting $Q_h^{k,T}$ to $\{q_h \in C^0(T) \mid q_h|_t \in \mathbb{P}_k(t) \quad \forall t \in \tau_h^T\} \subset H^1(T)$ and b_h^T to $b|_T$, where $\mathbb{P}_k(\omega)$ denotes the space of polynomials over $\omega \subseteq \Omega$ of order up to $k \in \mathbb{N}$; (ii) local nonconforming approximations by setting $Q_h^{k,T}$ to $\{q_h \in L^2(T) \mid q_h|_t \in \mathbb{P}_k(t) \quad \forall t \in \tau_h^T\} \subset L^2(T)$ and b_h^T to the following SWIPDG bilinear form: for $p, q \in Q_h^{k,T}$ and $\mu \in \mathscr{P}$, we define

$$b_h^T(p,q;\mu) := b^T(p,q;\mu) + \sum_{e \in \mathscr{F}_h^T} b_e(p,q;\mu),$$

with $b^T(p,q;\mu) := \int_T (\lambda(\mu)\kappa_\varepsilon \nabla p) \cdot \nabla q$, where $\mathscr{F}_h^T$ denotes the set of all inner faces of τ_h^T that share two elements $t^-, t^+ \in \tau_h^T$. The face bilinear form b_e for any inner or boundary face e of τ_h is given by $b_e(p,q;\mu) := b_c^e(q,p;\mu) + b_c^e(p,q;\mu) + b_p^e(p,q;\mu)$ with the coupling and penalty face bilinear forms b_c^e and b_p^e given by

$$b_c^e(p,q;\mu) := \int_e -\{\!\{\lambda(\mu)\kappa_\varepsilon \tilde{\Pi}\nabla p\}\!\}_e [\![q]\!]_e \quad \text{and} \quad b_p^e(p,q;\mu) := \int_e \sigma_e(\mu)[\![p]\!]_e [\![q]\!]_e,$$

respectively, with the L^2-orthogonal projection $\tilde{\Pi}$ from Definition 2. Given a function q which is two-valued on faces, its jump and weighted average are given by $[\![q]\!]_e := q^- - q^+$ and $\{\!\{q\}\!\}_e := \omega_e^- q^- + \omega_e^+ q^+$, respectively, on uniquely oriented inner faces $e = t^- \cap t^+$ for $t^\pm \in \tau_h$, and by $[\![q]\!]_e := \{\!\{q\}\!\}_e := q$ for boundary faces $e = t^- \cap \partial\Omega$, with the locally adaptive weights given by $\omega_e^- := \delta_e^+(\delta_e^+ + \delta_e^-)^{-1}$ and $\omega_e^+ := \delta_e^-(\delta_e^+ + \delta_e^-)^{-1}$, respectively, with $\delta_e^\pm := n_e \cdot \kappa_\varepsilon^\pm \cdot n_e$. Here, $n_e \in \mathbb{R}^d$ denotes the unique normal to a face e pointing away from t^- and $q^\pm := q|_{t^\pm}$. The positive penalty function is given by $\sigma_e(\mu) := \sigma h_e^{-1}\{\!\{\lambda(\mu)\}\!\}_e \sigma_\varepsilon^e$, where $\sigma \geq 1$ denotes a user-dependent parameter, $h_e > 0$ denotes the diameter of a face e, and the locally adaptive weight is given by $\sigma_\varepsilon^e := \delta_e^+\delta_e^-(\delta_e^+ + \delta_e^-)^{-1}$ on inner faces and by $\sigma_\varepsilon^e := \delta_e^-$ on boundary faces.

Given local approximations $Q_h^{k,T}$ and b_h^T on each subdomain $T \in \mathscr{T}_H$, we define the DG space by $Q_h^k := \oplus_{T\in\mathscr{T}_H} Q_h^{k,T}$ and couple the local discretizations along a coarse face $E \in \mathscr{F}_H$, by SWIPDG fluxes to obtain the global bilinear form $b : \mathscr{P} \to [Q_h^k \times Q_h^k \to \mathbb{R}]$, by

$$b(p, q; \mu) := \sum_{T\in\mathscr{T}_H} b_h^T(p, q; \mu) + \sum_{E\in\mathscr{F}_H} \sum_{e\in\mathscr{F}_h^E} b_e(p, q; \mu),$$

for $p, q \in Q_h^k$, $\mu \in \mathscr{P}$, where $\mathscr{F}_H$ denotes the set of all faces of the coarse grid $\mathscr{T}_H$ and where $\mathscr{F}_h^E$ denotes the set of fine faces of τ_h which lie on a coarse face $E \in \mathscr{F}_H$. Note that b is continuous and coercive with respect to Q_h^k in the DG norm $|||\cdot|||$. (see the next section) if the penalty parameter σ is chosen large enough (see [14] and the references therein concerning the choice of σ).

Depending on the choice of $\mathscr{T}_H$ and the local approximations, the above definition covers a wide range of discretizations, ranging from a standard conforming to a standard SWIPDG one; we refer to [14] for details. The semi-discrete problem for a single parameter then reads as (11.2) with $\tilde{Q} = Q_h^k$. Presuming $p_0 \in Q_h^k$ and using implicit Euler time stepping (compare Example 1) the fully-discrete problem reads: for each time step $n > 0$ find the DoF vector of $p_h^n(\mu) := p_h(n \cdot \Delta_t, \mu) \in Q_h^k$, denoted by $\underline{p_h^n(\mu)} \in \mathbb{R}^{\dim Q_h^k}$, such that

$$\Big(\underline{M_h} + \Delta_t\, \underline{b(\mu)}\Big)\underline{p_h^n(\mu)} = \Delta_t\, \underline{f_h} + \underline{p_h^{n-1}(\mu)}, \tag{11.8}$$

where $\underline{M_h}, \underline{b(\mu)} \in \mathbb{R}^{\dim Q_h^k \times \dim Q_h^k}$ and $\underline{f_h} \in \mathbb{R}^{\dim Q_h^k}$ denote the matrix and vector representations of $(\cdot,\cdot)_{L^2(\Omega)}$, $b(\cdot,\cdot;\mu)$ and $(f,\cdot)_{L^2(\Omega)}$, respectively, with respect to the basis of Q_h^k.

Model Reduction Let us assume that we are already given a reduced space $Q_{\text{red}} \subset Q_h^k$ (we postpone the discussion of how to find Q_{red} to Sect. 11.5). Given Q_{red}, we formally arrive at the reduced problem simply by Galerkin projection of (11.8) onto Q_{red}, just like traditional RB methods: for each time step $n > 0$ find the reduced DoF vector $\underline{p_{\text{red}}^n(\mu)} \in \mathbb{R}^{\dim Q_{\text{red}}}$, such that

$$\Big(\underline{M_{\text{red}}} + \Delta_t\, \underline{b_{\text{red}}(\mu)}\Big)\underline{p_{\text{red}}^n(\mu)} = \Delta_t\, \underline{f_{\text{red}}} + \underline{p_{\text{red}}^{n-1}(\mu)}, \tag{11.9}$$

with $p_{\text{red}}^0(\mu) := \Pi_{\text{red}}(p_0)$, where Π_{red} denotes the L^2-orthogonal projection onto Q_{red}, and where $\underline{M_{\text{red}}}, \underline{b_{\text{red}}(\mu)} \in \mathbb{R}^{\dim Q_{\text{red}} \times \dim Q_{\text{red}}}$ and $\underline{f_{\text{red}}} \in \mathbb{R}^{\dim Q_{\text{red}}}$ denote the matrix and vector representations of $(\cdot,\cdot)_{L^2(\Omega)}$, $b(\cdot,\cdot;\mu)$ and $(f,\cdot)_{L^2(\Omega)}$, respectively, with respect to the basis of Q_{red}.

As usual with RB methods, we can achieve an efficient offline/online splitting of the computational process by precomputing the restriction of the functionals and operators arising in (11.8) to Q_{red}, if those allow for an affine decomposition with respect to the parameter μ. For standard RB methods, where Q_{red} is spanned by reduced basis functions with global support, the matrix representation of the reduced L^2-product, for instance, would be given by $\underline{M_{\text{red}}} = \underline{\Pi_{\text{red}}} \cdot \underline{M_h} \cdot \underline{\Pi_{\text{red}}}^{\perp}$, where $\underline{\Pi_{\text{red}}} \in \mathbb{R}^{\dim Q_{\text{red}} \times \dim Q_h^k}$ denotes the matrix representation of Π_{red} (each row of $\underline{\Pi_{\text{red}}}$ corresponds to the DoF vector of one reduced basis function). For localized RB methods, however, we are given a local reduced basis on each subdomain $T \in \mathscr{T}_H$ (reflected in the structure of the reduced space, $Q_{\text{red}} = \oplus_{T \in \mathscr{T}_H} Q_{\text{red}}^T$) and all operators and functionals are localizable with respect to $\mathscr{T}_H$. Thus, the reduced basis projection can be carried out locally as well. For instance, since $(p, q)_{L^2(\Omega)} = \sum_{T \in \mathscr{T}_H} (p|_T, q|_T)_{L^2(T)}$, we locally obtain $\underline{M_{\text{red}}^T} = \underline{\Pi_{\text{red}}^T} \cdot \underline{M_h^T} \cdot \underline{\Pi_{\text{red}}^T}^{\perp} \in \mathbb{R}^{\dim Q_{\text{red}}^T \times \dim Q_{\text{red}}^T}$ for all $T \in \mathscr{T}_H$, where $\underline{\Pi_{\text{red}}^T} \in \mathbb{R}^{\dim Q_{\text{red}}^T \times \dim Q_h^{k,T}}$ and $\underline{M_h^T} \in \mathbb{R}^{\dim Q_h^{k,T} \times \dim Q_h^{k,T}}$ denote the matrix representations of the local L^2-orthogonal reduced basis projection and $(\cdot, \cdot)_{L^2(T)}$, respectively. The reduced L^2-product matrix $\underline{M_{\text{red}}} \in \mathbb{R}^{\dim Q_{\text{red}} \times \dim Q_{\text{red}}}$, with $\dim Q_{\text{red}} = \sum_{T \in \tau_h} \dim Q_{\text{red}}^T$, is then assembled by combining the local matrices using a standard DG mapping with respect to Q_{red}. In the same manner, the reduction of b can be carried out locally by projecting the local bilinear forms on each subdomain as well as the coupling bilinear forms with respect to all neighbors, yielding sparse reduced operators and products; we refer to [14] for details and implications.

11.4 Error Analysis

For our analysis we introduce the broken Sobolev space $H^1(\tau_h) := \{q \in L^2(\Omega) \mid q|_t \in H^1(t) \quad \forall t \in \tau_h\}$, containing $Q + \tilde{Q}$, since $Q_{\text{red}} \subset Q_h^k \subset H^1(\tau_h) \subset L^2(\Omega)$ and $H^1(\Omega) \subset H^1(\tau_h)$. Note that the domain of all operators, products and functionals of the previous section can be naturally extended to $H^1(\tau_h)$, for instance by using the broken gradient operator ∇_h, which is locally defined by $(\nabla_h q)|_t := \nabla(q|_t)$ for all $t \in \tau_h$. Using said operator in the definition of b^T, we define the parametric energy semi-norm (which is a norm only on $H_0^1(\Omega)$) by $|q|_\mu := \big(\sum_{T \in \mathscr{T}_H} b^T(q, q; \mu)\big)^{1/2}$ and the parametric DG norm by $|||q|||_\mu := \big(\sum_{T \in \mathscr{T}_H} b^T(q, q; \mu) + \sum_{e \in \mathscr{F}_h} b_p^e(q, q; \mu)\big)^{1/2}$, for $\mu \in \mathscr{P}$ and $q \in H^1(\tau_h)$, respectively, where $\mathscr{F}_h$ denotes the set of all faces of τ_h. Note that $|||q|||_\mu = |q|_\mu$ for $q \in H_0^1(\Omega)$.

Since we presume λ to be affinely decomposable with respect to μ, there exist $\Xi \in \mathbb{N}$ strictly positive coefficients $\theta_\xi : \mathscr{P} \to \mathbb{R}$ and nonparametric components $\lambda_\xi \in L^\infty(\Omega)$, such that $\lambda(\mu) = \sum_{\xi=1}^{\Xi} \theta_\xi(\mu)\lambda_\xi$. We can thus compare λ, and in particular $|||\cdot|||_\cdot$, for two parameters by means of $\alpha(\mu, \overline{\mu}) := \min_{\xi=1}^{\Xi} \theta_\xi(\mu)\theta_\xi(\overline{\mu})^{-1}$ and $\gamma(\mu, \overline{\mu}) := \max_{\xi=1}^{\Xi} \theta_\xi(\mu)\theta_\xi(\overline{\mu})^{-1}$:

$$\alpha(\mu, \overline{\mu})^{1/2}\, |||\cdot|||_{\overline{\mu}} \le |||\cdot|||_\mu \le \gamma(\mu, \overline{\mu})^{1/2}\, |||\cdot|||_{\overline{\mu}}. \tag{11.10}$$

Note that since we consider an energy norm here, usage of the above norm equivalence requires no additional offline computations, in contrast to the standard min-θ approach [15], where continuity and coercivity constants of $b(\cdot,\cdot;\overline{\mu})$ need to be computed when considering the H_0^1-norm. We also denote by $c_\varepsilon(\mu) > 0$ the minimum over $x \in \Omega$ of the smallest eigenvalue of the matrix $\lambda(x;\mu)\kappa_\varepsilon(x) \in \mathbb{R}^{d\times d}$.

We are interested in a fully computable and offline/online decomposable estimate on the full approximation error in a fixed energy norm. Therefore, we use the general framework presented in Sect. 11.2 and apply it to the parametric setting of the localized RB method. Since we use an implicit Euler time stepping for the reduced scheme we can readily apply Corollary 1 and Example 1 by specifying all arising terms.

Given any discontinuous function $q_{\mathrm{red}}(\mu) \in Q_{\mathrm{red}} \subset Q_h^k \not\subset H_0^1(\Omega)$, we use the Oswald interpolation operator $I_{\mathrm{OS}} : Q_h^k \to Q_h^k \cap H_0^1(\Omega)$, which consists of averaged evaluations of its source at Lagrange points of the grid τ_h (compare [14, Sect. 4] and the references therein), to compute the conforming and nonconforming parts of a function by $q_{\mathrm{red}}^c := I_{\mathrm{OS}}(q_{\mathrm{red}})$ and $q_{\mathrm{red}}^d := q_{\mathrm{red}} - q_{\mathrm{red}}^c$, respectively. Following Remark 1, we estimate the elliptic reconstruction error, $|||\varepsilon(n\cdot\Delta_t)|||_\mu$, by the localizable and offline/online decomposable a posteriori error estimate $\eta(p_{\mathrm{red}}(n\cdot\Delta_t;\mu);\mu,\mu,\tilde{\mu})$ from [14, Corollary 4.5], where $\tilde{\mu} \in \mathscr{P}$ denotes any fixed parameter.

Since b reduces on $H_0^1(\Omega)$ to the symmetric bilinear form (11.7), we have $|||b(\cdot,\cdot;\mu)|||_\mu = 1$ for any $\mu \in \mathscr{P}$. Denoting the Poincaré constant with respect to Ω by $C_P^\Omega > 0$, we can estimate $\|\Pi_{\mathrm{red}}(q)\|_{L^2(\Omega)} \le C_P^\Omega c_\varepsilon(\mu)^{-1}\,|||q|||_\mu$ for any $\mu \in \mathscr{P}$, $q \in H_0^1(\Omega)$ and $\left|\left|\left|\partial_t p_{\mathrm{red}}^d(t)\right|\right|\right|_{\mu,Q,-1} \le C_P^\Omega c_\varepsilon(\mu)^{-1}\left\|\partial_t p_{\mathrm{red}}^d(t)\right\|_{L^2(\Omega)}$ for $\partial_t p_{\mathrm{red}}^d \in L^2(0,T_{\mathrm{end}};Q_{\mathrm{red}})$ and $\mu \in \mathscr{P}$. We thus obtain the following estimate by applying Corollary 1 and Example 1 using the energy norm $|||\cdot|||_\mu$ and the norm equivalence (11.10).

Corollary 2 *Let the two partitions τ_h and $\mathscr{T}_H$ of Ω fulfill the requirements of [14, Theorem 4.2], namely: let τ_h be shape regular without hanging nodes and fine enough, such that all data functions can be assumed polynomial on each $t \in \tau_h$; let the subdomains $T \in \mathscr{T}_H$ be shaped, such that a local Poincaré inequality for functions in $H^1(T)$ with zero mean holds. For $\mu \in \mathscr{P}$ let $p(\cdot;\mu) \in L^2(0,T_{end};H_0^1(\Omega))$ denote the weak solution of the parabolic problem (11.1) and let $p_{\mathrm{red}}(\cdot;\mu) \in L^2(0,T_{end};Q_{\mathrm{red}})$ denote the reduced solution of the fully-discrete problem (11.9), where the constant function 1 is present in all local reduced bases spanning Q_{red}. It then holds for arbitrary $\hat{\mu},\overline{\mu},\tilde{\mu} \in \mathscr{P}$, that*

$$\begin{aligned}
&\|p(\mu) - p_{\mathrm{red}}(\mu)\|_{L^2(0,T_{end};|||\cdot|||_{\overline{\mu}})} \\
&\qquad \le \alpha(\mu,\overline{\mu})^{-1/2}\Big\{ \;\|e^c(0;\mu)\|_{L^2(\Omega)} + \sqrt{5}\,\left\|p_{\mathrm{red}}^d(\mu)\right\|_{L^2(0,T_{end};|||\cdot|||_\mu)} \\
&\qquad\qquad + 2\,\alpha(\mu,\hat{\mu})^{-1}\,C_{H,Q}^b(\hat{\mu})\,\left\|\partial_t p_{\mathrm{red}}^d(\mu)\right\|_{L^2(0,T_{end};L^2(\Omega))} \\
&\qquad\qquad + (\sqrt{5}+1)\,\eta_{\mathrm{ell.}}(p_{\mathrm{red}}(\mu),\mu,\tilde{\mu})
\end{aligned}$$

$$+\, 2\,\alpha(\mu,\hat{\mu})^{-1}\, C^b_{H,Q}(\hat{\mu})\, \|\mathscr{R}_T(p_{\mathrm{red}}(\mu);\mu)\|_{L^2(0,T_{end};L^2(\Omega))}\Big\}$$
$$=: \eta_{h,\mathrm{red}}(p_{\mathrm{red}}(\mu);\mu,\hat{\mu},\overline{\mu},\tilde{\mu})$$

with $C^b_{H,Q}(\hat{\mu}) = C^{\Omega}_P c_\varepsilon(\hat{\mu})^{-1}$ *and*

$$\eta_{\mathrm{ell.}}(p_{\mathrm{red}}(\mu);\mu,\tilde{\mu})^2 := \frac{4\Delta_t}{3}\sum_{n=0}^{n_t}\Big\{\eta_{\mathrm{OS2015}}(p_{\mathrm{red}}(n\cdot\Delta_t;\mu);\mu,\mu,\tilde{\mu})$$
$$+ \sum_{e\in\mathscr{F}_h} b^e_p(p_{\mathrm{red}}(n\cdot\Delta_t;\mu), p_{\mathrm{red}}(n\cdot\Delta_t;\mu);\mu)\Big\}$$

where η_{OS2015} *denotes the estimate* η *from [14, Corollary 4.5].*

In Corollary 2, we have the flexibility to choose three parameters $\hat{\mu},\overline{\mu},\tilde{\mu}\in\mathscr{P}$: the parameter $\overline{\mu}$ can be used to fix a norm throughout the computational process (for instance during the greedy basis generation), while the purpose of the parameters $\hat{\mu}$ and $\tilde{\mu}$ is to allow all quantities to be offline/online decomposable, cf. [14]. The price to pay for this flexibility are the additional occurrences of α, which are equal to 1 in the nonparametric case or if the parameters coincide.

11.5 Numerical Experiments

We consider (11.1) on $\Omega = [0,5]\times[0,1]$, $T_{\mathrm{end}} = 0.05$, with $p_0 = 0$ and the data functions f, κ and λ from the multiscale example in [14, Sect. 6.1]: κ_ε is the highly heterogeneous permeability tensor used in the first model of the 10th SPE Comparative Solution Project,[1] f models a source and two sinks and $\lambda(\mu) := 1 + (1-\mu)\lambda_c$, where λ_c models a high-conductivity channel. The role of the parameter $\mu\in\mathscr{P} := [0.1,1]$ is thus to toggle the existence of the channel, the maximum contrast of $\lambda(\mu)\kappa_\varepsilon$ amounts to 10^6 (compare Fig. 11.1).

Basis Generation On each subdomain $T\in\mathscr{T}_H$ we initialize the local reduced basis with φ^T_{red} :=`gram_schmidt`($\{1, f|_T\}$), where `gram_schmidt` denotes the Gram-Schmidt orthonormalization procedure (including re-orthonormalization for numerical stability) with respect to the full $H^1(T)$ product from our software package `pyMOR` (see below). The constant function 1 has to be present in the local reduced bases according to [14, Theorem 4.2] (to guarantee local mass conservation w.r.t. the subdomains), while the presence of f sharpens the a posteriori estimate by minimizing $\Pi_{\mathrm{red}}(f) - f$, as motivated by the elliptic reconstruction (11.3). We iteratively extend these initial bases using a variant of the POD-GREEDY algorithm [6]: in each iteration (i) the worst approximated parameter, say $\mu_*\in\mathscr{P}_{\mathrm{train}}$, is found

[1] http://www.spe.org/web/csp/index.html.

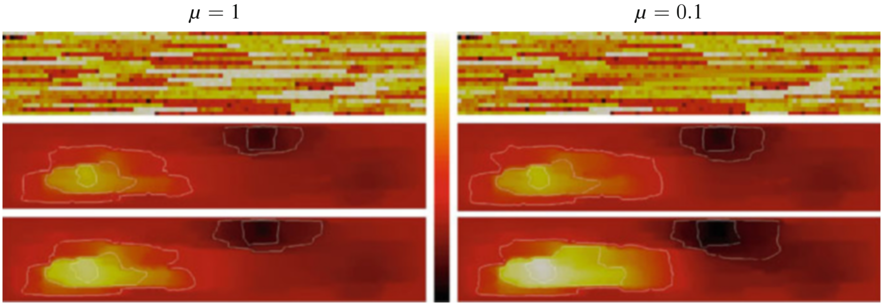


Fig. 11.1 Data functions and sample solutions on a grid with $|\tau_h| = 8000$ simplices for parameters $\mu = 1$ (*left column*) and $\mu = 0.1$ (*right column*). Both plots in the first row as well as the bottom four plots share the same color map (*middle*) with two different ranges. *First row*: logarithmic plot of $\lambda(\mu)\kappa_\varepsilon$ (dark: $1.41 \cdot 10^{-3}$, light: $1.41 \cdot 10^3$). Rest: plot of the pressure $p_h(t; \mu)$ (solution of (11.2), dark: $-3.92 \cdot 10^{-1}$, light: $7.61 \cdot 10^{-1}$, isolines at 10%, 20%, 45%, 75% and 95%) for $t = 0.01$ (middle row) and the end time $t = T_{\text{end}} = 0.05$ (bottom row). Note the presence of high-conductivity channels in the permeability (top left, light regions) throughout large parts of the domain. The parameter dependency models a removal of one such channel in the middle right of the domain

by evaluating the a posteriori error estimate from Corollary 2 over a set of training parameters $\mathscr{P}_{\text{train}} \subset \mathscr{P}$; (ii) a full solution trajectory $\{p_h(n \cdot \Delta_t; \mu_*) \,|\, 0 \leq n \leq n_t\}$ is computed using the discretization from Sect. 11.3; and (iii) the local reduced bases φ_{red}^T for each subdomain $T \in \mathscr{T}_H$ are extended by the dominant POD mode of the projection error of $\{p_h(n \cdot \Delta_t; \mu_*)|_T \,|\, 0 \leq n \leq n_t\}$, using the above Gram-Schmidt procedure.

Software Implementation We use the open-source Python software package pyMOR[2] [11] for all model reduction algorithms as well as for the time stepping. For the grids, operators, products and functionals we use the open-source C++ software package DUNE, in particular the generic discretization toolbox dune-gdt[3] (see [14, Sect. 6] and the references therein), compiled into a Python module to be directly usable in pyMOR's algorithms.

We use a simplicial triangulation for the fine grid τ_h, rectangular subdomains $T \in \mathscr{T}_H$ and 10 equally sized time steps for the implicit Euler scheme. Within each subdomain we use a local DG space of order 1, the resulting discretization thus coincides with the one proposed in [4].

We observe a comparable decay of the estimated error during the greedy basis generation in Fig. 11.2 for all subdomain configurations, though faster for a larger number of subdomains $|\mathscr{T}_H|$, where the reduced space is much richer. In particular, to reach the same prescribed error tolerance in the greedy algorithm, much less

[2] http://pymor.org.

[3] http://github.com/dune-community/dune-gdt.

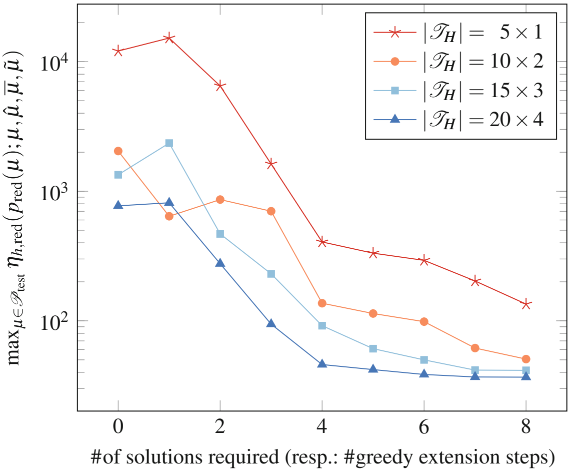


Fig. 11.2 Estimated error evolution during the POD-GREEDY basis generation for several subdomain configurations and $\hat{\mu} = \overline{\mu} = \tilde{\mu} = 0.1$, to minimize all occurrences of α in Corollary 2. Depicted is the maximum estimated error over a set of ten randomly chosen test parameters $\mathscr{P}_{\text{test}} \subset \mathscr{P}$ in each step of the greedy algorithm, which was configured to search over ten uniformly distributed training parameters

solution snapshots are required for larger numbers of subdomains. We refer to [12, Sect. 3.3] for a comparison of localized RB methods versus traditional RB methods.

11.6 Conclusion

In this contribution we used the elliptic reconstruction technique for a posteriori error estimation of parabolic problems [3, 5, 9, 10] to derive efficient and reliable true error control for the localized reduced basis method applied to scalar linear parabolic problems. Numerical experiments were given to demonstrate the applicability of the approach.

References

1. Albrecht, F., Haasdonk, B., Kaulmann, S., Ohlberger, M.: The localized reduced basis multiscale method. In: Proceedings of Algoritmy 2012, Conference on Scientific Computing, Vysoke Tatry, Podbanske, September 9–14, 2012, pp. 393–403. Slovak University of Technology in Bratislava, Publishing House of STU (2012)
2. Ali, M., Steih, K., Urban, K.: Reduced basis methods with adaptive snapshot computations. Adv. Comput. Math. **43**(2), 257–294 (2017). doi:10.1007/s10444-016-9485-9
3. Demlow, A., Lakkis, O., Makridakis, C.: A posteriori error estimates in the maximum norm for parabolic problems. SIAM J. Numer. Anal. **47**(3), 2157–2176 (2009). doi:10.1137/070708792
4. Ern, A., Stephansen, A.F., Zunino, P.: A discontinuous Galerkin method with weighted averages for advection–diffusion equations with locally small and anisotropic diffusivity. IMA J. Numer. Anal. **29**(2), 235–256 (2009)
5. Georgoulis, E.H., Lakkis, O., Virtanen, J.M.: A posteriori error control for discontinuous Galerkin methods for parabolic problems. SIAM J. Numer. Anal. **49**(2), 427–458 (2011). doi:10.1137/080722461

6. Haasdonk, B., Ohlberger, M.: Reduced basis method for finite volume approximations of parametrized linear evolution equations. M2AN Math. Model. Numer. Anal. **42**(2), 277–302 (2008). doi:10.1051/m2an:2008001
7. Hesthaven, J., Rozza, G., Stamm, B.: Certified reduced basis methods for parametrized partial differential equations. SpringerBriefs in Mathematics. Springer, Cham (2016). doi:10.1007/978-3-319-22470-1
8. Kaulmann, S., Flemisch, B., Haasdonk, B., Lie, K.A., Ohlberger, M.: The localized reduced basis multiscale method for two-phase flows in porous media. Int. J. Numer. Methods Eng. **102**(5), 1018–1040 (2015). doi:10.1002/nme.4773
9. Lakkis, O., Makridakis, C.: Elliptic reconstruction and a posteriori error estimates for fully discrete linear parabolic problems. Math. Comput. **75**(256), 1627–1658 (2006). doi:10.1090/S0025-5718-06-01858-8
10. Makridakis, C., Nochetto, R.H.: Elliptic reconstruction and a posteriori error estimates for parabolic problems. SIAM J. Numer. Anal. **41**(4), 1585–1594 (2003). doi:10.1137/S0036142902406314
11. Milk, R., Rave, S., Schindler, F.: pyMOR – generic algorithms and interfaces for model order reduction. SIAM J. Sci. Comput. **38**(5), S194–S216 (2016). doi:10.1137/15m1026614
12. Ohlberger, M., Rave, S., Schindler, F.: Model reduction for multiscale lithium-ion battery simulation. In: Karasözen, B., Manguoğlu, M., Tezer-Sezgin, M., Göktepe, S., Uğur, Ö. (eds.) Numerical Mathematics and Advanced Applications ENUMATH 2015, pp. 317–331. Springer International Publishing, Cham (2016). doi:10.1007/978-3-319-39929-4_31
13. Ohlberger, M., Schindler, F.: A-posteriori error estimates for the localized reduced basis multiscale method. In: Fuhrmann, J., Ohlberger, M., Rohde, C., (eds.) Finite Volumes for Complex Applications VII-Methods and Theoretical Aspects. Springer Proceedings in Mathematics & Statistics, vol. 77, pp. 421–429. Springer, Cham (2014). doi:10.1007/978-3-319-05684-5_41
14. Ohlberger, M., Schindler, F.: Error control for the localized reduced basis multiscale method with adaptive on-line enrichment. SIAM J. Sci. Comput. **37**(6), A2865–A2895 (2015). doi:10.1137/151003660
15. Patera, A.T., Rozza, G.: Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations, version 1.0. Technical Repotr, Copyright MIT 2006–2007, to appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering (2006)
16. Quarteroni, A., Manzoni, A., Negri, F.: Reduced Basis Methods for Partial Differential Equations. La Matematica per il 3+2. Springer, Cham (2016). doi:10.1007/978-3-319-15431-2
17. Verfürth, R.: A posteriori error estimation techniques for finite element methods. Numerical Mathematics and Scientific Computation. Oxford University Press, Oxford (2013). doi:10.1093/acprof:oso/9780199679423.001.0001
18. Yano, M.: A minimum-residual mixed reduced basis method: exact residual certification and simultaneous finite-element reduced-basis refinement. ESAIM: Math. Model. Numer. Anal. **50**, 163–185 (2015). doi:10.1051/m2an/2015039

Chapter 12
Efficient Reduction of PDEs Defined on Domains with Variable Shape

Andrea Manzoni and Federico Negri

Abstract In this work we propose a new, general and computationally cheap way to tackle parametrized PDEs defined on domains with variable shape when relying on the reduced basis method. We easily describe a domain by boundary parametrizations, and generate domain (and mesh) deformations by means of a solid extension, obtained by solving a linear elasticity problem. The proposed procedure is built over a two-stages reduction: (1) first, we construct a reduced basis approximation for the mesh motion problem; (2) then, we generate a reduced basis approximation of the state problem, relying on finite element snapshots evaluated over a set of reduced deformed configurations. A Galerkin-POD method is employed to construct both reduced problems, although this choice is not restrictive. To deal with unavoidable nonaffine parametric dependencies arising in both the mesh motion and the state problem, we apply a matrix version of the discrete empirical interpolation method, allowing to treat geometrical deformations in a non-intrusive, efficient and purely algebraic way. In order to assess the numerical performances of the proposed technique, we address the solution of a parametrized (direct) Helmholtz scattering problem where the parameters describe both the shape of the obstacle and other relevant physical features. Thanks to its easiness and efficiency, the methodology described in this work looks promising also in view of reducing more complex problems.

12.1 Introduction

The reduced basis (RB) method provides nowadays a very efficient approach for the numerical approximation of problems arising e.g. from engineering and applied sciences which require the repeated solution of differential equations. Well-known instances include partial differential equations (PDEs) depending on several parameters, PDE-constrained optimization, as well as optimal control and design problems. In all these cases, the RB method replaces the original large-

A. Manzoni (✉) • F. Negri
CMCS-MATHICSE-SB, Ecole Polytechnique Fédérale de Lausanne, Station 8, CH-1015 Lausanne, Switzerland
e-mail: andrea.manzoni@epfl.ch; federico.negri@epfl.ch

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_12

scale numerical problem (or high-fidelity approximation) originated by applying, e.g., a finite element (FE) method, with a reduced problem of substantially smaller dimension [14, 25].

In all these contexts, relevant instances of parametrized PDEs arise when dealing with problems defined over spatial domains undergoing geometrical transformations; this is the case of design problems, where being able to rapidly adapt existing meshes to design variations is essential to perform, e.g., shape optimization in an efficient way.

On the other hand, an offline/online stratagem, relying on the so-called *affine parametric dependence*, is required to gain a strong computational speedup when dealing with RB approximations to parametrized PDEs. In this respect, dealing with shape variations has often a major impact on the computational efficiency, since:

1. equipping the set of varying shape with a suitable *parametrization* is an involved, highly problem-dependent, task. In the RB context, parametric maps defined over the whole domain are needed to formulate the PDE problem on a parameter-independent reference configuration. However, in computed-aided design (CAD), *boundary parametrizations* under analytic form are usually defined for surfaces (in $d = 3$ dimensions) or curves ($d = 2$), rather than for the whole domain, thus preventing their direct use within the RB context;
2. geometrical parametrizations usually yield nonaffine parametric dependencies, so that an affine approximation of PDE operators has to be recovered through the *empirical interpolation method* (EIM) [3, 19] – or its discrete counterpart (DEIM) [6]. This usually entails an extensive work on the continuous formulation of the problem, as well as intrusive changes to its high-fidelity implementation.

Several techniques have been exploited to perform RB approximations of PDEs defined on varying domains. The simplest idea is to use *affine maps*, which induce an affine parametric dependence, but only enable elementary deformations [26].

More involved deformations can be obtained by introducing *nonaffine maps* yielding *volume-based parametrizations*. Within this class, we mention *free-form deformations* (FFD) [2, 17, 21, 27, 31] and interpolation relying on *radial basis functions* (RBF) [9, 10, 20, 23] as remarkable instances. Both techniques originate global deformations by combining the displacements of a set of control points. FFD deal with a cartesian lattice of control points and a tensor product of splines to combine control points displacements; these latter are instead interpolated in the RBF case, where the control points can be freely located inside the domain. In both cases, however, selecting the number of control points, their position and admissible displacements is far from being trivial.

Another option relies instead on the use of *transfinite mappings*, which define the interior points of the original domain as linear combinations of points on the boundaries [11]. In particular, each edge of the original domain is obtained as a one-to-one mapping of the corresponding edge on the reference domain, through a vector of geometrical parameters, see e.g. [8, 15, 16].

Finally, the use of B-splines and NURBS basis functions as a possible tool to define more complex and realistic geometrical parametrizations has also been explored in [22, 28], in combination with isogeometric analysis for the generation of the high-fidelity approximation.

The mesh motion strategy considered in this work allows to simplify the way to deal with geometrical deformations, by relying on (1) simple boundary parametrizations, and (2) the solution of a solid extension problem. Moreover, we exploit a recently proposed matrix version of DEIM (MDEIM, [5, 23, 37]) to perform inexpensive evaluations of the online matrix operators for both the deformation and the state problem. Hence, we first recover an affine parametric dependence in the high-fidelity arrays appearing in both problems, by applying MDEIM and DEIM for matrix and vector operators, respectively. This is performed in a purely algebraic, black-box, in order to overcome the application of the EIM on the continuous formulation of the problem, which is usually highly demanding, see e.g. [4, 12, 24]. Then, we perform the RB approximation of both the deformation and the state problem, relying on a Galerkin-POD technique.

The paper is structured as follows. In Sect. 12.2 we describe the proposed mesh deformation technique. In Sect. 12.3 we introduce the class of problems we deal with in this work, as well as the main features of the RB approximation framework we develop. The whole computational procedure is then applied in Sect. 12.4 for the sake of the efficient solution of a parametrized (direct) Helmholtz scattering problem. Finally, some conclusions are reported in Sect. 12.5.

12.2 Solid Extension Mesh Moving Techniques

Let $\widetilde{\Omega} \subset \mathbb{R}^d$ be a spatial domain with boundary $\widetilde{\Gamma}$, where $d = 2, 3$ is the number of space dimensions. We denote by $\widetilde{\Gamma}_h$ a discretization of the boundary $\widetilde{\Gamma}$ and by $\widetilde{\Omega}_h$ a volumetric mesh of that geometry, e.g. a triangular mesh in 2D or a tetrahedral mesh in 3D. Given a boundary deformation $\widetilde{\Gamma}_h \mapsto \Gamma_h$, mesh deformation techniques adapt the mesh $\widetilde{\Omega}_h$ such that *(i)* the updated mesh Ω_h conforms to the updated boundary, i.e. $\partial\Omega_h = \Gamma_h$ and *(ii)* the geometric embedding of Ω_h (i.e., its nodes positions) is modified while keeping fixed the mesh topology (i.e., its connectivity).

Among a wide range of existing mesh deformation techniques, here we focus on the so called *mesh-based variational methods* (see, e.g., [32]). These latter compute smooth harmonic [1], biharmonic [13] or elastic [33–35] deformations by solving Laplacian, bi-Laplacian or elasticity problems, respectively. Specifically, we consider this latter, which is often referred to as solid-extension mesh moving technique (SEMMT) [33–35].

Before describing the method, let us first introduce a non-overlapping decomposition of the boundary Γ into a deformable portion $\widetilde{\gamma}$ and a fixed one $\partial\Omega \setminus \widetilde{\gamma}$. Given a boundary displacement $\boldsymbol{h} \in [H^{1/2}(\widetilde{\gamma})]^d$ such that $\gamma = \{\mathbf{x} \in \mathbb{R}^d \ : \ \mathbf{x} = \tilde{\mathbf{x}} + \boldsymbol{h},\ \tilde{\mathbf{x}} \in \widetilde{\Omega}\}$, the SEMMT generates a deformed domain Ω as

$$\Omega(\boldsymbol{h}) = \{\mathbf{x} \in \mathbb{R}^d \ : \ \mathbf{x} = \tilde{\mathbf{x}} + \boldsymbol{d}(\boldsymbol{h}),\ \tilde{\mathbf{x}} \in \widetilde{\Omega}\}$$

where $\boldsymbol{d} = \boldsymbol{d}(\boldsymbol{h}) \in [H^1(\widetilde{\Omega})]^d$ is the displacement field solution of the following linear elasticity problem:

$$\begin{aligned} -\mathrm{div}(\boldsymbol{\sigma}(\boldsymbol{d})) &= \mathbf{0} && \text{in } \widetilde{\Omega} \\ \boldsymbol{d} &= \boldsymbol{h} && \text{on } \widetilde{\gamma} \\ \boldsymbol{d} &= \mathbf{0} && \text{on } \partial\widetilde{\Omega} \setminus \widetilde{\gamma}. \end{aligned} \tag{12.1}$$

Here, $\boldsymbol{\sigma}(\boldsymbol{d}) = 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda \mathrm{div}(\boldsymbol{d})\boldsymbol{I}$ is the Cauchy stress tensor , λ and μ are the Lamé constants, $\boldsymbol{I}$ is the identity tensor and $\boldsymbol{\varepsilon}(\boldsymbol{d}) = \frac{1}{2}(\nabla\boldsymbol{d} + \nabla\boldsymbol{d}^T)$ is the strain tensor.

Since the SEMMT is then applied at the discrete level, we introduce the FE trial and test functions spaces

$$V_h(\boldsymbol{h}) = \{\boldsymbol{v}_h \,|\, \boldsymbol{v}_h \in [P_h]^d,\ \boldsymbol{v}_h|_{\widetilde{\gamma}_h} = \boldsymbol{h},\ \boldsymbol{v}_h|_{\partial\widetilde{\Omega}_h \setminus \widetilde{\gamma}_h} = \mathbf{0}\},$$
$$V_h^0 = \{\boldsymbol{v}_h \,|\, \boldsymbol{v}_h \in [P_h]^d,\ \boldsymbol{v}_h|_{\partial\widetilde{\Omega}} = \mathbf{0}\},$$

where P_h denotes a FE space made of piecewise polynomial nodal basis functions. The high-fidelity FE approximation of (12.1) reads as follows: find $\boldsymbol{d}_h \in V_h(\boldsymbol{h})$ such that

$$\int_{\widetilde{\Omega}_h} \boldsymbol{\sigma}(\boldsymbol{d}_h) : \boldsymbol{\varepsilon}(\boldsymbol{v}_h)\, d\widetilde{\Omega} = 0 \qquad \forall \boldsymbol{v}_h \in V_h^0. \tag{12.2}$$

As described in [34], the method is then augmented with a proper Jacobian-based stiffening in order to enhance the mesh quality. To this end, the way we account for the Jacobian of the transformation from the element domain to the physical domain is altered by replacing the global integrals in (12.2) as follows

$$\int_{\widetilde{\Omega}_h} [\cdots]\, d\widetilde{\Omega} = \sum_{e \in \widetilde{\Omega}_h} \int_{\Xi} [\cdots]^e J^e \, d\Xi \quad \longrightarrow \quad \sum_{e \in \widetilde{\Omega}_h} \int_{\Xi} [\cdots]^e J^e \left(\frac{J^0}{J^e}\right)^{\eta} d\Xi. \tag{12.3}$$

Here, Ξ denotes the reference element, J^e is the Jacobian of the element e, J^0 is an arbitrary scaling parameter and $\eta \in \mathbb{R}_+$ is the so-called stiffening power.

At the algebraic level, problem (12.2) yields a linear system of large dimension $N_h^d \times N_h^d$ to be solved,

$$\mathbb{B}(\eta)\mathbf{d}_h = \mathbf{g}(\boldsymbol{h}), \tag{12.4}$$

where $\mathbf{d}_h \in \mathbb{R}^{N_h^d}$, $\mathbb{B}(\eta) \in \mathbb{R}^{N_h^d \times N_h^d}$ and the right-hand side vector $\mathbf{g}(\boldsymbol{h}) \in \mathbb{R}^{N_h^d}$ encodes the action of the nonhomogeneous Dirichlet condition imposed on $\widetilde{\gamma}$.

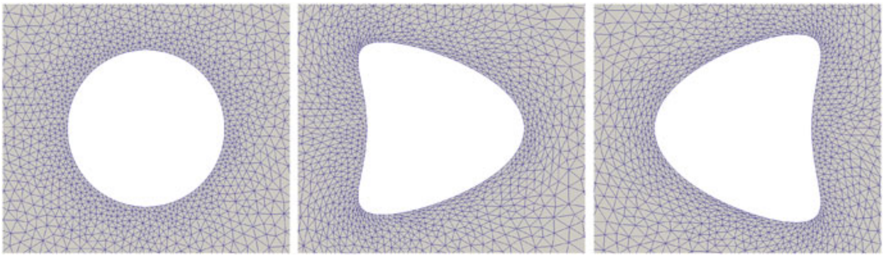


Fig. 12.1 *Left*: undeformed volumetric mesh $\widetilde{\Omega}_h$; *center* and *right*: examples of deformed volumetric meshes $\Omega_h(\boldsymbol{h})$ obtained through (12.5)

Given a boundary displacement $\boldsymbol{h}$, solving (12.4) thus allows to obtain a deformed volumetric mesh

$$\Omega_h(\boldsymbol{h}) = \{\mathbf{x}_h \in \mathbb{R}^d \ : \ \mathbf{x}_h = \tilde{\mathbf{x}}_h + \mathbf{d}_h(\eta, \boldsymbol{h}), \ \tilde{\mathbf{x}}_h \in \widetilde{\Omega}_h\}.$$

which satisfies the requirements *(i)* and *(ii)*.

The boundary displacement $\boldsymbol{h}$ can be generated in different ways depending on the application at hand. In this work, we consider the simplest case where $\boldsymbol{h}$ is given in the form of a parameter-dependent analytic function. To make an example, let us consider the 2D domain $\Omega = D(\mathbf{0}; 5) \setminus D(\mathbf{0}; 1)$, where $D(\mathbf{x}_c; r)$ denotes the open disk of center $\mathbf{x}_c$ and radius r. A family of boundary deformations parametrized with respect to a vector of two parameters (α, β) could be defined as follows [7]

$$\boldsymbol{h} = [\cos(t) + \alpha \cos(2t) - \alpha, \ \beta \sin(t)] \tag{12.5}$$

with $t = \text{atan2}(\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2) \in [0, 2\pi]$, see Fig. 12.1. A more advanced alternative would be to use suitable boundary morphing techniques like RBF or the design element approach, see e.g. [38]. In all these cases, however, the boundary deformation is controlled through a (possibly small) set $\mathscr{P}_g \subset \mathbb{R}^g$ of geometrical parameters $\boldsymbol{\mu}_g$, i.e. $\boldsymbol{h} = \boldsymbol{h}(\boldsymbol{\mu}_g)$. For instance, in the case of expression (12.5) we have $\boldsymbol{\mu}_g = (\alpha, \beta, \eta)$, by considering the stiffening power $\eta \in \mathbb{R}_+$ as a further parameter. As a result, the *mesh motion (MM) problem* turns into the following parametrized problem: given $\boldsymbol{\mu}_g \in \mathscr{P}_g$, find the displacement field $\mathbf{d}_h \in \mathbb{R}^{N_h^d}$ such that

$$\mathbb{B}(\boldsymbol{\mu}_g)\mathbf{d}_h = \mathbf{g}(\boldsymbol{\mu}_g). \tag{12.6}$$

Note that the dependence of problem (12.6) on the parameters defining the family of boundary deformations is only through its right-hand side $\mathbf{g}$.

12.3 Reduced Basis Approximation: POD-Galerkin Techniques and Matrix DEIM

Solving problem (12.1) allows to compute a displacement field over the whole domain; any parameter-dependent instance of the domain thus result by applying the displacement to the reference, parameter-independent, domain $\widetilde{\Omega} \subset \mathbb{R}^d$, that is,

$$\Omega(\boldsymbol{\mu}_g) = \{\mathbf{x} \in \mathbb{R}^d \ : \ \mathbf{x} = \tilde{\mathbf{x}} + \mathbf{d}(\boldsymbol{\mu}_g),\ \tilde{\mathbf{x}} \in \widetilde{\Omega}\}, \quad \boldsymbol{\mu}_g \in \mathscr{P}_g;$$

the computational mesh $\widetilde{\Omega}_h$ over which the state problem is solved is then given by

$$\Omega_h(\boldsymbol{\mu}_g) = \{\mathbf{x}_h \in \mathbb{R}^d \ : \ \mathbf{x}_h = \tilde{\mathbf{x}}_h + \mathbf{d}_h(\boldsymbol{\mu}_g),\ \tilde{\mathbf{x}}_h \in \widetilde{\Omega}_h\}, \quad \boldsymbol{\mu}_g \in \mathscr{P}_g,$$

being $\mathbf{d}_h = \mathbf{d}_h(\boldsymbol{\mu}_g)$ the solution of the high-fidelity problem (12.6). Since its solution for any parameter vector $\boldsymbol{\mu}_g \in \mathscr{P}_g$ would be computationally expensive, we rather approximate the displacement field by relying on the RB method, whose main ingredients will be detailed in the following. Hence, we approximate $\mathbf{d}_h(\boldsymbol{\mu}_g) \approx \mathbb{V}\mathbf{d}_N(\boldsymbol{\mu}_g)$ as a linear combination of N_d (deformation) basis functions, being $\mathbf{d}_N \in \mathbb{R}^{N_d}$ a vector of coefficients. The latter is the solution of a problem obtained by projecting the high-fidelity system (12.6) onto the basis $\mathbb{V}$, as we shall describe later.

As a result, the set of parametrized domains we deal with is given by

$$\left\{\Omega_h^N(\boldsymbol{\mu}_g) = \{\mathbf{x}_h^N \in \mathbb{R}^d \ : \ \mathbf{x}_h^N = \tilde{\mathbf{x}}_h + \mathbb{V}\mathbf{d}_N(\boldsymbol{\mu}_g),\ \tilde{\mathbf{x}}_h \in \widetilde{\Omega}_h\}, \quad \boldsymbol{\mu}_g \in \mathscr{P}_g\right\}; \tag{12.7}$$

provided that the error $\|\mathbf{d}_h(\boldsymbol{\mu}_g) - \mathbb{V}\mathbf{d}_N(\boldsymbol{\mu}_g)\|$ is sufficiently small—this is indeed ensured by standard algorithms in the RB context—$\Omega_h^N(\boldsymbol{\mu}_g)$ yields an accurate approximation of $\Omega_h(\boldsymbol{\mu}_g)$.

12.3.1 Formulation of the State Problem

Let us now move to the state problem we finally want to solve. For the sake of illustration, we consider as state problem the case of a scalar linear elliptic stationary PDE, although the proposed technique can be extended to more general problems in a straightforward way. Let us denote by $W = W(\boldsymbol{\mu}_g)$ a suitable Hilbert space, defined over the parameter-dependent domain $\Omega(\boldsymbol{\mu}_g) \subset \mathbb{R}^d$; in abstract form, the parametrized problem we focus on can be written as follows: given $\boldsymbol{\mu} = (\boldsymbol{\mu}_g, \boldsymbol{\mu}_p) \in \mathscr{P} = \mathscr{P}_g \times \mathscr{P}_p \subset \mathbb{R}^{g+p}$, find $u(\boldsymbol{\mu}) \in W(\boldsymbol{\mu}_g)$ such that

$$a(u(\boldsymbol{\mu}), v; \boldsymbol{d}(\boldsymbol{\mu}_g), \boldsymbol{\mu}_p) = f(v; \boldsymbol{d}(\boldsymbol{\mu}_g), \boldsymbol{\mu}_p) \qquad \forall v \in W(\boldsymbol{\mu}_g); \tag{12.8}$$

here $\boldsymbol{\mu}_p$ denotes a vector of physical parameters only affecting the state problem. Note that the presence of the displacement $\boldsymbol{d}(\boldsymbol{\mu}_g)$, playing the role of known parametrized field in (12.8), induces a dependence of both the bilinear form $a(\cdot,\cdot;\boldsymbol{d}(\boldsymbol{\mu}_g),\boldsymbol{\mu}_p) : W(\boldsymbol{\mu}_g) \times W(\boldsymbol{\mu}_g) \to \mathbb{C}$ and the linear form $f(\cdot;\boldsymbol{d}(\boldsymbol{\mu}_g),\boldsymbol{\mu}_p) : W(\boldsymbol{\mu}_g) \to \mathbb{C}$ on $\boldsymbol{\mu}_g$, too.

Here we assume that $a(\cdot,\cdot;\boldsymbol{d}(\boldsymbol{\mu}_g),\boldsymbol{\mu}_p)$ is continuous and weakly coercive over $W \times W$, and that $f(\cdot;\boldsymbol{d}(\boldsymbol{\mu}_g),\boldsymbol{\mu}_p)$ is continuous, for any $(\boldsymbol{\mu}_g,\boldsymbol{\mu}_p)$, so that problem (12.8) admits a unique solution thanks to Nečas theorem.

The high-fidelity, FE approximation of problem (12.8) can then be obtained upon defining a FE space $W_h(\boldsymbol{\mu}_g) \subset W(\boldsymbol{\mu}_g)$ over the domain $\Omega_h^N(\boldsymbol{\mu}_g)$, and seeking $u_h(\boldsymbol{\mu}) \in W_h(\boldsymbol{\mu}_g)$ such that

$$a(u_h(\boldsymbol{\mu}), v_h; \mathbf{d}_N(\boldsymbol{\mu}_g), \boldsymbol{\mu}_p) = f(v_h; \mathbf{d}_N(\boldsymbol{\mu}_g), \boldsymbol{\mu}_p) \qquad \forall v_h \in W_h(\boldsymbol{\mu}_g); \tag{12.9}$$

note that we have already considered the RB approximation of the displacement field. From an algebraic standpoint, problem (12.9) yields a linear system of large dimension $N_h^u \times N_h^u$ to be solved,

$$\mathbb{A}(\boldsymbol{\mu})\mathbf{u}_h(\boldsymbol{\mu}) = \mathbf{f}(\boldsymbol{\mu}), \tag{12.10}$$

where $\mathbb{A}(\boldsymbol{\mu}) = \mathbb{A}(\mathbf{d}_N(\boldsymbol{\mu}_g);\boldsymbol{\mu}_p) \in \mathbb{R}^{N_h^u \times N_h^u}$ and $\mathbf{f}(\boldsymbol{\mu}) = \mathbf{f}(\mathbf{d}_N(\boldsymbol{\mu}_g);\boldsymbol{\mu}_p) \in \mathbb{R}^{N_h^u}$.

12.3.2 POD-Galerkin Reduced Order Models (ROMs)

Problems (12.6) and (12.10) share the same nature of parameter-dependent, high dimensional, linear systems arising from the discretization of two different second-order parametrized PDEs. To solve them efficiently, we rely in both cases on the RB method, thus approximating the unknowns $\mathbf{u}_h$ in a basis $\mathbb{W} \in \mathbb{R}^{N_h^u \times N_u}$, $\mathbf{d}_h$ in a basis $\mathbb{V} \in \mathbb{R}^{N_h^d \times N_d}$ of reduced dimensions $N_u \ll N_h^u$, $N_d \ll N_h^d$, i.e. $\mathbf{u}_h(\boldsymbol{\mu}) \approx \mathbb{W}\mathbf{u}_N(\boldsymbol{\mu})$, $\mathbf{d}_h(\boldsymbol{\mu}_g) \approx \mathbb{V}\mathbf{d}_N(\boldsymbol{\mu}_g)$. Then, we enforce the orthogonality of the residual of each equation to $\mathbb{W}$ and $\mathbb{V}$, respectively, thus resulting in two Galerkin-RB problems under the following form: given $\boldsymbol{\mu}_g \in \mathscr{P}_g$, find $\mathbf{d}_N(\boldsymbol{\mu}_g) \in \mathbb{R}^{N_d}$

$$\mathbb{B}_N(\boldsymbol{\mu}_g)\mathbf{d}_N(\boldsymbol{\mu}_g) = \mathbf{g}_N(\boldsymbol{\mu}_g), \tag{12.11}$$

and then, given $\boldsymbol{\mu}_p \in \mathscr{P}_p$, find $\mathbf{u}_N(\boldsymbol{\mu}) \in \mathbb{R}^{N_u}$ such that

$$\mathbb{A}_N(\boldsymbol{\mu})\mathbf{u}_N(\boldsymbol{\mu}) = \mathbf{f}_N(\boldsymbol{\mu}), \tag{12.12}$$

where the reduced matrices and vectors are given by

$$\mathbb{B}_N(\boldsymbol{\mu}_g) = \mathbb{V}^T\mathbb{B}(\boldsymbol{\mu}_g)\mathbb{V}, \qquad \mathbf{g}_N(\boldsymbol{\mu}_g) = \mathbb{V}^T\mathbf{g}(\boldsymbol{\mu}_g), \tag{12.13}$$

$$\mathbb{A}_N(\boldsymbol{\mu}) = \mathbb{W}^T\mathbb{A}(\mathbf{d}_N(\boldsymbol{\mu}_g);\boldsymbol{\mu}_p)\mathbb{W}, \qquad \mathbf{f}_N(\boldsymbol{\mu}) = \mathbb{W}^T\mathbf{f}(\mathbf{d}_N(\boldsymbol{\mu}_g);\boldsymbol{\mu}_p). \tag{12.14}$$

Here, we rely on the proper orthogonal decomposition (POD) method for the construction of the RB spaces. Once a set of snapshots of problems (12.6) and (12.10) has been computed, the singular value decomposition of the corresponding correlation matrices automatically yield optimal sets of orthonormal basis functions; see, e.g., [25] for further details. Note that each snapshot of problem (12.10) is computed on a different spatial domain, depending on the value of $\boldsymbol{\mu}_g$; nevertheless, this is not a concern, since we have assumed that the mesh deformation induced by $\mathbf{d}_h(\boldsymbol{\mu}_g)$ (and, correspondingly, by $\mathbf{d}_N(\boldsymbol{\mu}_g)$) does not affect the mesh connectivity and, as a result, the connectivity graph of the matrix $\mathbb{A}(\boldsymbol{\mu})$, too. The resulting POD-Galerkin technique allows to obtain two problems (12.11) and (12.12) of very small dimension.

Assembling the reduced matrices and vectors as in (12.13) and (12.14) when $\boldsymbol{\mu} \in \mathscr{P}$ varies is still too expensive in order to achieve efficient offline construction and online evaluation of the RB problem. As already mentioned, if the system matrices (resp. vectors) can be expressed as an affine combination of constant matrices (resp. vectors) weighted by suitable parameter-dependent coefficients, each term of the weighted sums can be projected offline onto the RB space spanned by $\mathbb{W}$, $\mathbb{V}$, respectively. For instance, if we assume that the matrix $\mathbb{A}(\boldsymbol{\mu})$ admits an affine decomposition

$$\mathbb{A}(\boldsymbol{\mu}) = \sum_{q=1}^{M_A} \theta_q^A(\boldsymbol{\mu})\mathbb{A}_q, \tag{12.15}$$

then

$$\mathbb{A}_N(\boldsymbol{\mu}) = \mathbb{W}^T\mathbb{A}(\boldsymbol{\mu})\mathbb{W} = \sum_{q=1}^{M_A} \theta_q^A(\boldsymbol{\mu})\, \mathbb{W}^T\mathbb{A}_q\mathbb{W},$$

where $\theta_q^A \colon \mathscr{P} \mapsto \mathbb{R}$ and $\mathbb{A}_q \in \mathbb{R}^{N_h^u \times N_h^u}$ are given functions and matrices, respectively, for $q = 1, \dots, M_A$; a similar affine decomposition made by M_f terms is required for the vector $\mathbf{f}(\boldsymbol{\mu})$ as well. Since the reduced matrices $\mathbb{W}^T\mathbb{A}_q\mathbb{W} \in \mathbb{R}^{N_u \times N_u}$ can be precomputed and stored offline, the online construction of the RB arrays in (12.14) for a given $\boldsymbol{\mu}$ is fast and efficient as long as $M_A, M_f \ll N_h^u$; a similar conclusion clearly holds for the RB arrays in (12.13), too.

In order to recover the affine structure (12.15) in those cases where the operator $\mathbb{A}(\boldsymbol{\mu})$ is nonaffine (i.e., (12.15) is not readily available), we must introduce a further level of reduction, called *hyper-reduction*; we thus refer to the resulting ROM as

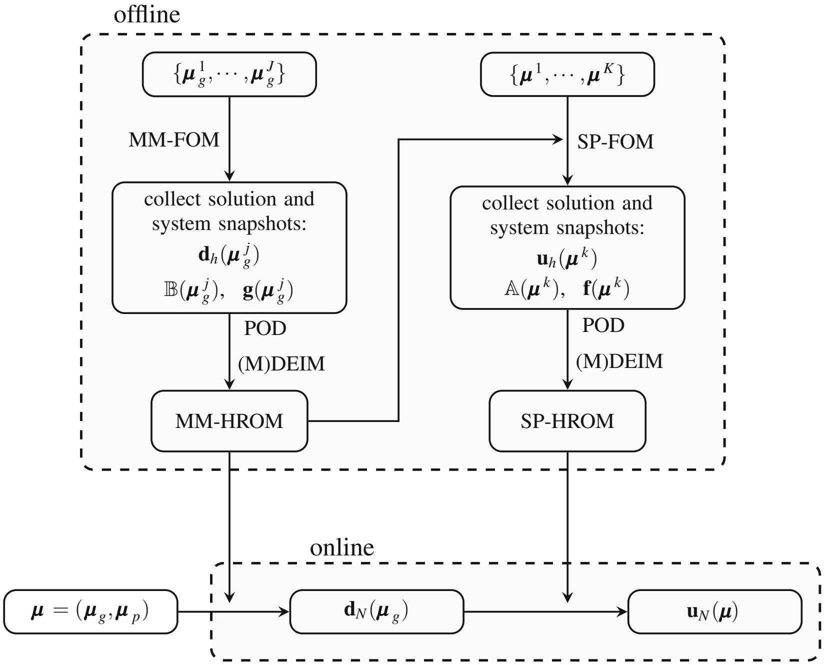

offline
$\{\boldsymbol{\mu}_g^1, \cdots, \boldsymbol{\mu}_g^J\}$
MM-FOM
collect solution and system snapshots:
$\mathbf{d}_h(\boldsymbol{\mu}_g^j)$
$\mathbb{B}(\boldsymbol{\mu}_g^j)$, $\mathbf{g}(\boldsymbol{\mu}_g^j)$
POD
(M)DEIM
MM-HROM
$\{\boldsymbol{\mu}^1, \cdots, \boldsymbol{\mu}^K\}$
SP-FOM
collect solution and system snapshots:
$\mathbf{u}_h(\boldsymbol{\mu}^k)$
$\mathbb{A}(\boldsymbol{\mu}^k)$, $\mathbf{f}(\boldsymbol{\mu}^k)$
POD
(M)DEIM
SP-HROM
online
$\boldsymbol{\mu} = (\boldsymbol{\mu}_g, \boldsymbol{\mu}_p)$
$\mathbf{d}_N(\boldsymbol{\mu}_g)$
$\mathbf{u}_N(\boldsymbol{\mu})$

Fig. 12.2 Scheme of offline and online phases for the geometry and state reduction. Here MM-FOM and MM-HROM refer to (12.6) and (12.11), respectively (i.e. the full and hyper-reduced order models for the mesh motion (MM) problem). SP-FOM and SP-HROM refer instead to (12.10) and (12.12), respectively (i.e. the full and hyper-reduced order models for the state problem (SP))

hyper-ROM (HROM). Here, we rely on DEIM to approximate the vectors $\mathbf{f}(\boldsymbol{\mu})$ and $\mathbf{g}(\boldsymbol{\mu}_g)$, and its matrix variant MDEIM to approximate $\mathbb{A}(\boldsymbol{\mu})$ and $\mathbb{B}(\boldsymbol{\mu}_g)$. A schematic summary of the entire offline/online computational strategy is offered in Fig. 12.2.

12.3.3 Matrix DEIM

For the sake of space—and because of its relative novelty—here we only detail the way DEIM can be used to approximate a parameter-dependent matrix $\mathbb{K}(\tau)\colon \mathscr{T} \mapsto \mathbb{R}^{N_h \times N_h}$, where τ denotes a generic parameters vector. Given $\mathbb{K}(\tau)\colon \mathscr{T} \mapsto \mathbb{R}^{N_h \times N_h}$, the problem is to find $M \ll N_h$ functions $\theta_q\colon \mathscr{T} \mapsto \mathbb{R}$ and parameter-independent matrices $\mathbb{K}_q \in \mathbb{R}^{N_h \times N_h}$, $1 \leq q \leq M$, such that

$$\mathbb{K}(\tau) \approx \mathbb{K}_m(\tau) = \sum_{q=1}^{M} \theta_q(\tau)\, \mathbb{K}_q. \tag{12.16}$$

The offline stage of this procedure consists of two main steps. First we express $\mathbb{K}(\tau)$ in vector format by stacking its columns, that is, we set $\mathbf{k}(\tau) = \mathrm{vec}(\mathbb{K}(\tau)) \in \mathbb{R}^{N_h^2}$. Hence, (12.16) can be reformulated as: find $\{\boldsymbol{\Phi}, \boldsymbol{\theta}(\tau)\}$ such that

$$\mathbf{k}(\tau) \approx \mathbf{k}_m(\tau) = \boldsymbol{\Phi}\boldsymbol{\theta}(\tau), \tag{12.17}$$

where $\boldsymbol{\Phi} \in \mathbb{R}^{N_h^2 \times M}$ is a τ-independent basis and $\boldsymbol{\theta}(\tau) \in \mathbb{R}^M$ the corresponding coefficients vector. Then, we apply DEIM as in [6] to a set of snapshots $\boldsymbol{\Lambda} = [\mathrm{vec}(\mathbb{K}(\tau_1)), \ldots, \mathrm{vec}(\mathbb{K}(\tau_{n_s}))]$ in order to obtain the basis $\boldsymbol{\Phi}$ and a set of interpolation indices $\mathscr{I} \subset \{1, \cdots, N_h^2\}$. The former is computed by applying the POD technique over the columns of $\boldsymbol{\Lambda}$, whereas the latter is iteratively selected by employing the *magic points* algorithm [19].

During the online phase, given a new $\tau \in \mathscr{T}$, we can compute $\mathbb{K}_m(\tau)$ as

$$\mathbb{K}_m(\tau) = \mathrm{vec}^{-1}(\boldsymbol{\Phi}\boldsymbol{\theta}(\tau)) \qquad \text{with} \qquad \boldsymbol{\theta}(\tau) = \boldsymbol{\Phi}_{\mathscr{I}}^{-1}\mathbb{K}_{\mathscr{I}}(\tau), \tag{12.18}$$

where $\boldsymbol{\Phi}_{\mathscr{I}}$ and $\mathbb{K}_{\mathscr{I}}(\tau)$ denote the matrices formed by the $\mathscr{I}$ rows of $\boldsymbol{\Phi}$ and $\mathbb{K}(\tau)$, respectively. We point out that, for the sake of model order reduction, the crucial step in the online evaluation of $\mathbb{K}_m(\tau)$ is the computation of $\mathbb{K}_{\mathscr{I}}(\tau)$. Nevertheless, this operation can be performed efficiently when $\mathbb{K}(\tau)$ results from a FE discretization of a PDE operator, by employing the same assembly routine used for the high-fidelity problem on the *reduced mesh* associated to the selected interpolation indices; see, e.g., [23] for further details.

12.4 Numerical Example

As a proof of concept of the proposed technique, we consider a (direct) scattering problem dealing with the Helmholtz equation. Scattering problems are meant to study the effect that a bounded obstacle (or *scatterer*) has on incident waves, depending on the geometrical properties of the body; these latter are considered as geometrical parameters of interest. Such a problem is relevant in a wide range of applications such as the design of sonars and radars, medical imaging, geophysical exploration, and nondestructive testing [7]. Given the incident wave, the goal of a direct scattering problem is to determine the scattered wave for the known obstacle; from a numerical standpoint, this is also a premise in view of the (indeed, very challenging) inverse scattering problem, in which the obstacle shape has to be reconstructed from far-field measurements. Helmholtz equations have already been tackled by RB methods, see, e.g. [18, 29, 30].

Let $B \subset \mathbb{R}^d$, $d = 2, 3$, the domain of an object with boundary $\gamma = \partial B$, and assume that $\gamma = \gamma(\boldsymbol{\mu}_g)$ is parametrized with respect to a vector of geometrical parameters $\boldsymbol{\mu}_g \in \mathscr{P}_g \subset \mathbb{R}^g$. The exterior domain is defined by the unbounded region $\mathscr{R} = \mathbb{R}^d \setminus B$; here we restrict ourselves to the case $d = 2$, although everything can

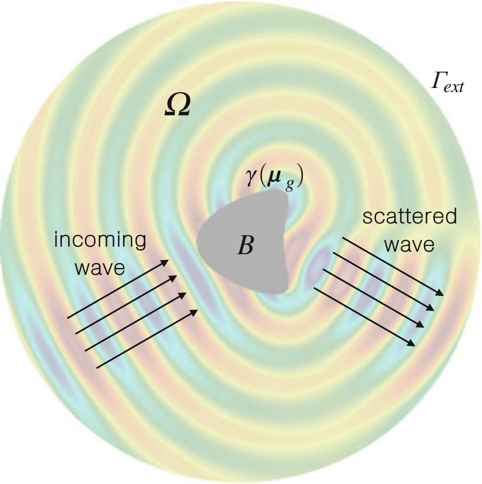


Fig. 12.3 Sketch of the domain and of the underlying physics; background coloring given by $\Re(u)$ for $\boldsymbol{\mu}_g = (-0.4, 1.2, 1.2)$, $\kappa = 5, a = \pi/6$

be simply extended to the case $d = 3$ as well. Instead of considering an exterior acoustics problem in an unbounded domain, we truncate this latter by an artificial boundary Γ_{ext} where local absorbing boundary conditions are imposed; as a result, we deal with a bounded computational domain Ω, see Fig. 12.3.

Here we consider the propagation of time harmonic waves, for which the acoustic pressure P can be separated as $P(\mathbf{x}, t; \boldsymbol{\mu}) = \Re(u(\mathbf{x}; \boldsymbol{\mu})e^{-i\omega t})$; the complex amplitude $u = u(\mathbf{x}; \boldsymbol{\mu})$ then satisfies the Helmholtz equation

$$\begin{aligned} \Delta u + \kappa^2 u = 0 & \qquad \text{in } \Omega(\boldsymbol{\mu}_g) \\ u = -e^{i\kappa\, \mathbf{a}\cdot\mathbf{x}} & \qquad \text{on } \gamma(\boldsymbol{\mu}_g) \\ \nabla u \cdot \mathbf{n} - i\kappa u = 0 & \qquad \text{on } \Gamma_{ext}. \end{aligned} \tag{12.19}$$

Here $\kappa = \omega/c$ is the wave number, $\omega = 2\pi f$ the angular frequency and $c = 340$ cm s^{-1} the speed of sound. The scattered wave is time-harmonic, but not necessary plane, whereas the incident wave is considered to be a plane, time-harmonic wave $u^i(\mathbf{x}, t; \boldsymbol{\mu}) = e^{i(\kappa \mathbf{a}\cdot\mathbf{x} - \omega t)}$ with amplitude $A = 1$ and direction $\mathbf{a} = (\cos(a), \sin(a))^T$. On the boundary Γ_{ext} we prescribe a *first-order absorbing boundary condition*, yielding an approximation of the *Sommerfeld radiation condition*

$$\lim_{r\to\infty} r\left(\frac{\partial u}{\partial r} - iku\right) = 0$$

usually imposed in the case of unbounded domains [36]. In addition to the geometrical parameters $\boldsymbol{\mu}_g = (\alpha, \beta, \eta)$ encoding the shape of the obstacle B and the stiffening power, we also consider a vector of physical parameters $\boldsymbol{\mu}_p = (\kappa, a) \in$

$\mathscr{P}_p \subset \mathbb{R}^p$ to model different scenarios where the wave number, as well as the direction, of the incident wave can vary.

We now apply the methodology developed in the previous sections to problem (12.19). The full-order model for the state Helmholtz problem is given by a $\mathbb{P}_1^{bubble}$ finite element approximation of (12.19), yielding a linear system of dimension $N_h^u = 147,272$ obtained using a mesh made of 97,934 triangular elements. Similarly, the full-order model for the mesh motion problem (12.1) is built using $\mathbb{P}_1^{bubble}$ finite elements, yielding a linear system of dimension $N_h^d = 294,544$. Concerning the parameter range, we select $\alpha \in [-1/2, 1/2]$, $\beta \in [-0.81.2]$, $\eta \in [0, 1.4]$; instead, physical parameters range in $\kappa \in [2, 5]$, $a \in [0, \pi/6]$.

We first consider the reduction of the mesh motion problem. We begin by computing a set of 50 solution, matrix and vector snapshots corresponding to 50 parameter samples selected by Latin Hypercube Sampling (LHS) design in $\mathscr{P}_g$. The eigenvalues of the correlation matrices of matrix and vector snapshots are reported in Fig. 12.4. Based on their decay, we retain the first $M_B = 5$ and $M_g = 11$ terms and then perform MDEIM and DEIM, respectively, to select the sets of interpolation indices. The reduced basis $\mathbb{V}$ is instead obtained by extracting $N = 10$ POD modes. In all these cases, a tolerance of $\varepsilon_{POD} = 10^{-5}$ has been imposed on the relative information content criterion, see [25]. Very few basis functions are then required to build an accurate ROM for the mesh motion problem, as it is shown by the convergence analysis reported in Fig. 12.5, left. The magnitude of the resulting RB displacement $\mathbf{d}_N(\boldsymbol{\mu}))$ is instead reported in Fig. 12.5, right, for different parameter values.

Then, we turn to the reduction of the Helmholtz state problem—where $\Omega(\boldsymbol{\mu}_g)$ is approximated by $\Omega_h^N(\boldsymbol{\mu}_g)$, see Eq. (12.7)—following the same procedure as above. Regarding the system approximation, in this case $M_A = 60$ and $M_f = 142$ terms are selected out of 400 matrix and vector snapshots; concerning state reduction, we retain $N = 120$ basis functions, see Fig. 12.6.

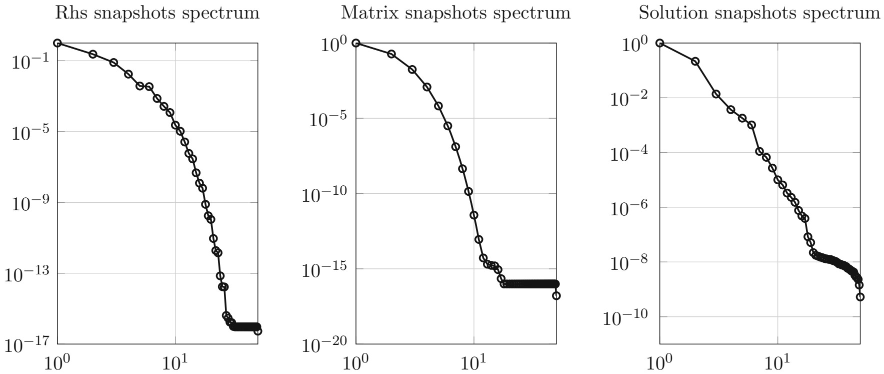


Fig. 12.4 Decay of the singular values of system and solution snapshots for the mesh motion problem

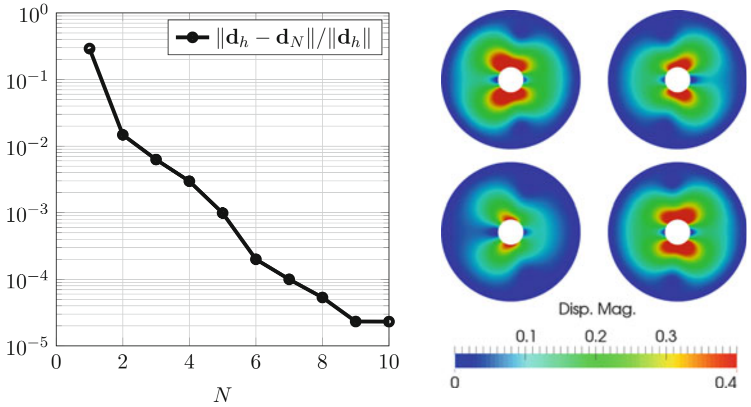


Fig. 12.5 *Left*: Relative error on the solution of the mesh motion problem (averaged over a test sample of 100 parameter values). *Right*: magnitude of the displacement $\mathbf{d}_N(\boldsymbol{\mu})$ for different parameter values

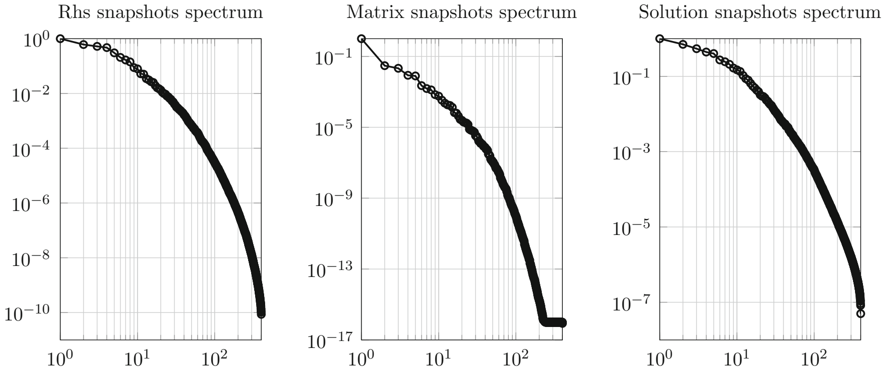


Fig. 12.6 Decay of the singular values of system and solution snapshots for the Helmholtz problem

Note that the decay of the eigenvalues of the correlation matrix is much slower than in the previous case highlighting a stronger variability of $\mathbf{u}_h(\boldsymbol{\mu})$—as well as of the problem arrays $\mathbb{A}(\boldsymbol{\mu})$, $\mathbf{f}(\boldsymbol{\mu})$—with respect to combined variations of both geometrical and physical parameters. This also translates into a much slower error convergence with respect to the RB dimension N_u, see Fig. 12.7, right. The resulting reduced mesh (see Fig. 12.7, left) is made of 291 elements, corresponding to about the 0.3% of the original ones; note that they concentrate around the obstacle, i.e. in the region where problem sensitivity to shape variations is higher. Some instances of the solution $u_N(\boldsymbol{\mu})$ to the Helmholtz equation obtained with the proposed technique are finally shown in Fig. 12.8; as a concluding remark, we point out that the online solution of both the reduced mesh motion (12.11) and the reduced Helmholtz

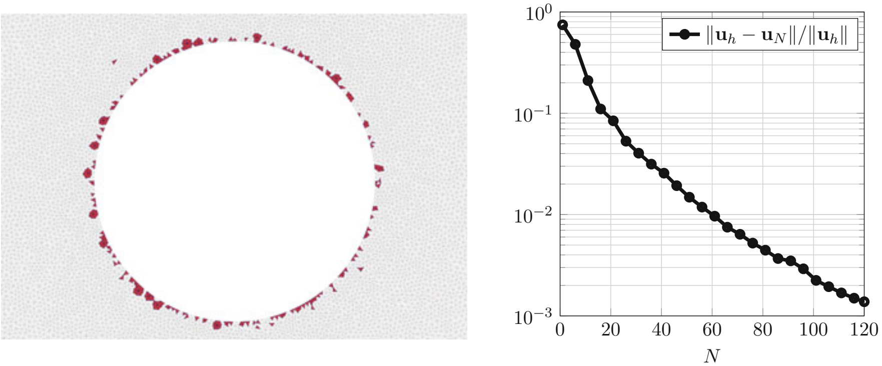


Fig. 12.7 *Left*: zoom of the reduced mesh (*red* elements) around the object. *Right*: relative error on the solution of the Helmholtz problem (averaged over a testing set of 200 parameter values)

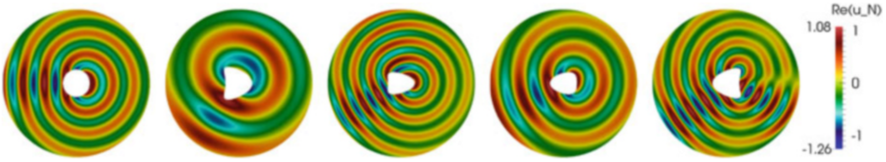


Fig. 12.8 $\Re(u_N(\boldsymbol{\mu}))$ for different values of the parameters

problem (12.12) problems takes about $0.26s$, thus realizing a computational speedup of about 25 times with respect to the finite element FOM.[1]

12.5 Conclusions

In this work we have presented a general and automatic way to deal with the efficient solution of parameterized PDEs defined on domains with variable shape. This framework combines a mesh motion technique relying on the solution of a solid extension problem, a POD-Galerkin reduced basis method, and a further hyper-reduction stage based on DEIM/MDEIM techniques. Compared to already existing strategies for handling mesh deformations in the RB context, the technique exploited in this work allows to directly define global domain deformations starting from boundary parametrizations. Indeed, relying on boundary—rather than volume—parametrizations is a

[1] These CPU times refer to computations performed on a workstation with Intel Core i5-2400S CPU and 16 GB of RAM. The implementation of the mentioned algorithms has been done using the `redbKIT` library (http://redbkit.github.io/redbKIT/), developed in the MATLAB®environment and distributed under BSD 2-clause license.

very common way to handle shape variations when design optimization is performed.

Although the proposed framework has been tested on a simplified case, where a single explicit boundary parametrization has been considered, its capabilities look promising in view of tackling a wider range of problems. For instance, it allows to handle different types of parametrizations simultaneously, each one defined on a separate portion of the boundary. It is also well-suited for three-dimensional problems, provided a suitable boundary parametrization is defined in order to describe those portions of the domain undergoing shape changes, and for time-dependent problems, provided the deformation of the domain is described as a function of time and parameters. The application of the proposed technique to nonlinear problems is also straightforward, since the generation of the ROM for the solid extension problem is independent from the one required by the state problem. Furthermore, it also applies in the case where a database of boundary deformations results from the solution of a different problem—rather than from an explicit formula. This is the case, e.g., of fluid-structure interaction problems, where the deformation of the fluid-structure interface is an unknown of the problem itself. In this case, the ROM for the solid extension problem and the one for the fluid flow both rely upon snapshots of the full-order fluid-structure interaction problem solved at the offline stage; further work is ongoing in this respect.

References

1. Baker, T.: Mesh movement and metamorphosis. Eng. Comput. **18**(3), 188–198 (2002)
2. Ballarin, F., Manzoni, A., Rozza, G., Salsa, S.: Shape optimization by free-form deformation: existence results and numerical solution for Stokes flows. J. Sci. Comput. **60**(3), 537–563 (2014)
3. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. C. R. Math. Acad. Sci. Paris **339**(9), 667–672 (2004)
4. Canuto, C., Tonn, T., Urban, K.: A posteriori error analysis of the reduced basis method for non-affine parameterized nonlinear PDEs. SIAM J. Numer. Anal. **47**(3), 2001–2022 (2009)
5. Carlberg, K., Tuminaro, R., Boggs, P.: Preserving Lagrangian structure in nonlinear model reduction with application to structural dynamics. SIAM J. Sci. Comput. **37**(2), B153–B184 (2015)
6. Chaturantabut, S., Sorensen, D.C.: Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. **32**(5), 2737–2764 (2010)
7. Colton, D., Kress, R.: Inverse Acoustic and Electromagnetic Scattering Theory. Springer, Berlin (2012)
8. Deparis, S., Løvgren, A.E.: Stabilized reduced basis approximation of incompressible three-dimensional Navier-Stokes equations in parametrized deformed domains. J. Sci. Comput. **50**(1), 198–212 (2012)
9. Deparis, S., Forti, D., Quarteroni, A.: A rescaled localized radial basis function interpolation on non-cartesian and nonconforming grids. SIAM J. Sci. Comput. **36**(6), A2745–A2762 (2014)
10. Forti, D., Rozza, G.: Efficient geometrical parametrisation techniques of interfaces for reduced-order modelling: application to fluid-structure interaction coupling problems. Int. J. Comput. Fluid. Dyn. **28**(3–4), 158–169 (2014)

11. Gordon, W., Hall, C.: Construction of curvilinear co-ordinate systems and applications to mesh generation. Int. J. Numer. Methods Eng. **7**(4), 461–477 (1973)
12. Grepl, M.A., Maday, Y., Nguyen, N.C., Patera, A.T.: Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. ESAIM Math. Model. Numer. Anal. **41**(3), 575–605 (2007)
13. Helenbrook, B.: Mesh deformation using the biharmonic operator. Int. J. Numer. Methods Eng. **56**(7), 1007–1021 (2003)
14. Hesthaven, J., Rozza, G., Stamm, B.: Certified Reduced Basis Methods for Parametrized Partial Differential Equations. SpringerBriefs in Mathematics. Springer, Switzerland (2016)
15. Iapichino, L., Quarteroni, A., Rozza, G.: A reduced basis hybrid method for the coupling of parametrized domains represented by fluidic networks. Comput. Methods Appl. Mech. Eng. **221–222**, 63–82 (2012)
16. Jäggli, C., Iapichino, L., Rozza, G.: An improvement on geometrical parameterizations by transfinite maps. C. R. Acad. Sci. Paris. Sér. I **352**(3), 263–268 (2014)
17. Lassila, T., Rozza, G.: Parametric free-form shape design with PDE models and reduced basis method. Comput. Methods Appl. Mech. Eng. **199**(23–24), 1583–1592 (2010)
18. Lassila, T., Manzoni, A., Rozza, G.: On the approximation of stability factors for general parametrized partial differential equations with a two-level affine decomposition. ESAIM Math. Model. Numer. Anal. **46**(6), 1555–1576 (2012)
19. Maday, Y., Nguyen, N.C., Patera, A.T., Pau, G.S.H.: A general multipurpose interpolation procedure: the magic points. Commun. Pure Appl. Anal. **8**(1), 383–404 (2009)
20. Manzoni, A., Quarteroni, A., Rozza, G.: Model reduction techniques for fast blood flow simulation in parametrized geometries. Int. J. Numer. Methods Biomed. Eng. **28**(6–7), 604–625 (2012)
21. Manzoni, A., Quarteroni, A., Rozza, G.: Shape optimization of cardiovascular geometries by reduced basis methods and free-form deformation techniques. Int. J. Numer. Methods Fluids **70**(5), 646–670 (2012)
22. Manzoni, A., Salmoiraghi, F., Heltai, L.: Reduced basis isogeometric methods (RB-IGA) for the real-time simulation of potential flows about parametrized NACA airfoils. Comput. Methods Appl. Mech. Eng. **284**, 1147–1180 (2015)
23. Negri, F., Manzoni, A., Amsallem, D.: Efficient model reduction of parametrized systems by matrix discrete empirical interpolation. J. Comput. Phys. **303**, 431–454 (2015)
24. Nguyen, N.C.: A posteriori error estimation and basis adaptivity for reduced-basis approximation of nonaffine-parametrized linear elliptic partial differential equations. J. Comput. Phys. **227**, 983–1006 (2007)
25. Quarteroni, A., Manzoni, A., Negri, F.: Reduced Basis Methods for Partial Differential Equations. An Introduction. Unitext, vol. 92. Springer, Switzerland (2016)
26. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. Arch. Comput. Methods Eng. **15**, 229–275 (2008)
27. Rozza, G., Lassila, T., Manzoni, A.: Reduced basis approximation for shape optimization in thermal flows with a parametrized polynomial geometric map. In: Hesthaven, J.S., Rønquist, E. (eds.) Spectral and High Order Methods for Partial Differential Equations. Selected papers from the ICOSAHOM '09 conference, June 22–26, Trondheim, Norway. Lecture Notes in Computational Science and Engineering, vol. 76, pp. 307–315. Springer, Berlin/Heidelberg (2011)
28. Salmoiraghi, F., Ballarin, F., Heltai, L., Rozza, G.: Isogeometric analysis-based reduced order modelling for incompressible linear viscous flows in parametrized shapes. Adv. Model. Simul. Eng. Sci **3**(1), 21 (2016)
29. Sen, S.: Reduced basis approximation and a posteriori error estimation for non-coercive elliptic problems: application to acoustics. Ph.D. thesis, Massachusetts Institute of Technology (2007)
30. Sen, S., Veroy, K., Huynh, D.B.P., Deparis, S., Nguyen, N.C., Patera, A.T.: "Natural norm" a posteriori error estimators for reduced basis approximations. J. Comput. Phys. **217**(1), 37–62 (2006)

31. Sieger, D., Botsch, M., Menzel, S.: On shape deformation techniques for simulation-based design optimization. In: Perotto, S., Formaggia, L. (eds.) New Challenges in Grid Generation and Adaptivity for Scientific Computing. SEMA SIMAI Springer Series, vol. 5, pp. 281–303. Springer, Switzerland (2015)
32. Staten, M., Owen, S., Shontz, S., Salinger, A., Coffey, T.: A comparison of mesh morphing methods for 3D shape optimization. In: Proceedings of the 20th International Meshing Roundtable, pp. 293–311. Springer (2011)
33. Stein, K., Tezduyar, T., Benney, R.: Mesh moving techniques for fluid-structure interactions with large displacements. J. Appl. Mech. **70**(1), 58–63 (2003)
34. Stein, K., Tezduyar, T., Benney, R.: Automatic mesh update with the solid-extension mesh moving technique. Comput. Methods Appl. Mech. Eng. **193**(21), 2019–2032 (2004)
35. Tezduyar, T., Behr, M., Mittal, S., Johnson, A.: Computation of unsteady incompressible flows with the stabilized finite element methods: Space-time formulations, iterative strategies and massively parallel implementations. In: New Methods in Transient Analysis, vol. 246/AMD, pp. 7–24. ASME, New York (1992)
36. Thompson, L.: A review of finite-element methods for time-harmonic acoustics. J. Acoust. Soc. Am. **119**(3), 1315–1330 (2006)
37. Wirtz, D., Sorensen, D.C., Haasdonk, B.: A posteriori error estimation for DEIM reduced nonlinear dynamical systems. SIAM J. Sci. Comput. **36**(2), A311–A338 (2014)
38. Zahr, M.J., Farhat, C.: Progressive construction of a parametric reduced-order model for PDE-constrained optimization. Int. J. Numer. Methods Eng. **102**(5), 1111–1135 (2015)

Chapter 13
Localized Reduced Basis Approximation of a Nonlinear Finite Volume Battery Model with Resolved Electrode Geometry

Mario Ohlberger and Stephan Rave

Abstract In this contribution we present first results towards localized model order reduction for spatially resolved, three-dimensional lithium-ion battery models. We introduce a localized reduced basis scheme based on non-conforming local approximation spaces stemming from a finite volume discretization of the analytical model and localized empirical operator interpolation for the approximation of the model's nonlinearities. Numerical examples are provided indicating the feasibility of our approach.

13.1 Introduction

Over the recent years, three dimensional lithium (Li) ion battery models that fully resolve the microscopic geometry of the battery electrodes have become a subject of active research in electrochemistry [10]. These models are also studied in the collaborative research project MULTIBAT, where the influence of the microscopic electrode geometry plays in important role in understanding the degradation process of Li-plating [9].

Due to the strongly nonlinear character of these models and the large number of degrees of freedom of their discretization, numerical simulations are time consuming and parameter studies quickly turn prohibitively expensive. Our work in context of the MULTIBAT project has shown that model reduction techniques such as reduced basis (RB) methods are able to vastly reduce the computational complexity of parametrized microscale battery models while retaining the full microscale features of their solutions [12, 15]. Still, such methods depend on the solution of the full high-dimensional model for selected parameters during the so-called offline phase. When only relatively few simulations of the model are required—as it is typically the case for electrochemistry simulations where one

M. Ohlberger • S. Rave (✉)
Applied Mathematics Muenster & Center for Nonlinear Science, University of Muenster, Einsteinstr. 62, 48149 Muenster, Germany
e-mail: mario.ohlberger@uni-muenster.de; stephan.rave@uni-muenster.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_13

is mainly interested in the qualitative behaviour of the battery cell—the offline phase can quickly take nearly as much time as the simulation of the full model for all parameters of interest would have required. It is, therefore, paramount to reduce the number of full model solves as much possible. Localized RB methods construct spatially localized approximation spaces from few global model solves or even by only solving adequate local problems (see also [5, 6, 13] and the references therein). Thus, these methods are a natural choice for accelerating the offline phase of RB schemes, in particular for problems with a strong microscale character such as geometrically resolved electrochemistry simulations.

While localized RB methods have been studied extensively for linear problems and while there are first results for instationary problems [13, 15], we are not aware of any previous work treating nonlinear models. In this contribution, we introduce a localized RB scheme for nonlinear finite volume battery models, which builds local approximation and interpolation spaces by decomposition of global solution snapshots w.r.t. a given coarse triangulation of the computational domain (Sect. 13.5). As a preparation, we will first briefly review the microscale battery model under consideration (Sect. 13.2), its discretization (Sect. 13.3) and finally its RB approximation (Sect. 13.4). We will close with first numerical experiments that investigate the applicability of localized RB techniques to the problem at hand (Sect. 13.6).

13.2 Analytical Model

As in [12, 15], we consider the microscale battery model introduced in [10] (without taking thermal effects into account and assuming constant t_+). In this model, the battery cell is described via coupled systems of partial differential equations for the concentration of Li$^+$-ions and the electrical potential ϕ for each part of the cell: the electrolyte, positive and negative electrode, as well as positive and negative current collector (Fig. 13.1).

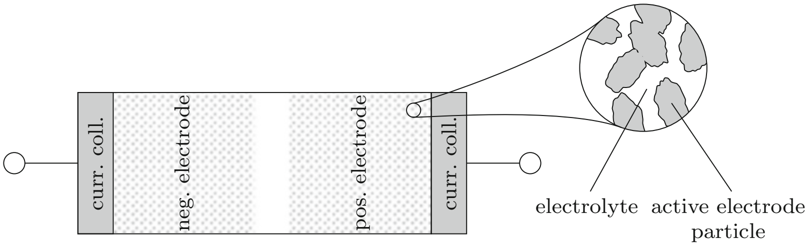


Fig. 13.1 Sketch of a lithium-ion battery cell. The cell is connected via two metallic current collectors which are in contact with the negative/positive cell electrodes. The porous electrodes are composed of active electrode particles into which Li-ions intercalate from the electrolyte filling the pore space of the electrodes

In the electrolyte, the change of the concentration c is governed by a diffusion process, whereas ϕ is determined by a stationary potential equation with source term depending non-linearly on c:

$$\begin{aligned}\frac{\partial c}{\partial t} - \nabla \cdot (D_e \nabla c) &= 0,\\ -\nabla \cdot \left(\kappa \frac{1-t_+}{F} RT \frac{1}{c} \nabla c - \kappa \nabla \phi\right) &= 0.\end{aligned}$$

In the electrodes, the evolution of c, i.e. the intercalation of Li-ions into the active particles, is again driven by diffusion. The potential ϕ no longer depends on the Li-ion distribution:

$$\begin{aligned}\frac{\partial c}{\partial t} - \nabla \cdot (D_s \nabla c) &= 0,\\ -\nabla \cdot (\sigma \nabla \phi) &= 0.\end{aligned}$$

No Li-ions enter the metallic current collectors, so $c \equiv 0$ on this part of the domain, whereas ϕ is again given as:

$$-\nabla \cdot (\sigma \nabla \phi) = 0.$$

The reaction at the interface between active electrode particles and the electrolyte is governed by the so-called Butler-Volmer kinetics which determine the electric current $j = \nabla \phi \cdot n$ from the active particle into the electrolyte as

$$j = 2k\sqrt{c_e c_s (c_{max} - c_s)} \sinh\left(\frac{\phi_s - \phi_e - U_0(\frac{c_s}{c_{max}})}{2RT} \cdot F\right), \tag{13.1}$$

where c_e, ϕ_e (c_s, ϕ_s) are the concentration and potential on the electrolyte (solid particle) side of the interface. The Li-ion flux N over the interface proportionally depends on j and is given by $N = j/F$. Note that the Butler-Volmer relations ensure the coupling between both considered variables and, through the exponential dependence on the overpotential $\phi_s - \phi_e - U_0(c_s/c_{max})$, lead to a highly nonlinear behaviour of the system.

Finally, continuity conditions for ϕ are imposed between electrode and current collector, whereas there is no coupling between electrolyte and current collector. The following boundary conditions are imposed: $\phi = U_0(c(0)/c_{max})$ at the negative current collector boundary, Neumann boundary conditions at the positive current collector (applied fixed charge/discharge rate) and periodic boundary conditions for c and ϕ at the remaining domain boundaries. We denote the initial concentration at time $t = 0$ by $c_0 = c(0)$, the final time is denoted as T. All appearing natural/material constants as well as the initial data is summarised in Table 13.1.

Table 13.1 Constants used in the battery model

Symbol	Unit	Value	Description
D_e	$\frac{cm^2}{s}$	$1.622 \cdot 10^{-6}$	Collective interdiffusion coefficient in electrolyte
D_s	$\frac{cm^2}{s}$	10^{-10}	Ion diffusion coefficient in electrodes
σ	$\frac{s}{cm}$	10	Electronic conductivity in neg. electrode
		0.38	Electronic conductivity in pos. electrode
κ	$\frac{s}{cm}$	0.02	Ion conductivity
c_{max}	$\frac{mol}{cm^3}$	$24681 \cdot 10^{-6}$	Maximum Li$^+$ concentration in neg. electrode
		$23671 \cdot 10^{-6}$	Maximum Li$^+$ concentration in pos. electrode
k	$\frac{Acm^{2.5}}{mol^{1.5}}$	0.002	Reaction rate at neg. electrode/electrolyte interface
		0.2	Reaction rate at pos. electrode/electrolyte interface
c_0	$\frac{mol}{cm^3}$	$1200 \cdot 10^{-6}$	Initial concentration in electrolyte
		$2639 \cdot 10^{-6}$	Initial concentration in neg. electrode
		$20574 \cdot 10^{-6}$	Initial concentration in pos. electrode
t_+		0.39989	Transference number
T	K	298	Temperature
F	$\frac{As}{mol}$	96, 487	Faraday constant
R	$\frac{J}{mol\,K}$	8.314	Universal gas constant

The open circuit potential U_0 for a state of charge s is give as $U_0(s) = (-0.132 + 1.41 \cdot e^{-3.52s}) \cdot V$ for the negative electrode and as $U_0(s) = [0.0677504 \cdot \tanh(-21.8502 \cdot s + 12.8268) - 0.105734 \cdot \big((1.00167 - s)^{-0.379571} - 1.576\big) - 0.045 \cdot e^{-71.69 \cdot s^8} + 0.01 \cdot e^{-200 \cdot (s-0.19)} + 4.06279] \cdot V$ for the positive electrode

13.3 Finite Volume Discretization

Following [16], the continuous model is discretized using a basic cell centered finite volume scheme on a voxel grid. Each voxel is assigned a unique subdomain and the Butler-Volmer conditions are chosen as numerical flux on grid faces separating an electrolyte from an electrode voxel. We obtain a single nonlinear finite volume operator $A_\mu : V_h \oplus V_h \to V_h \oplus V_h$ for the whole computational domain, where V_h denotes the space of piecewise constant grid functions and μ indicates a parameter we want to vary. In the following, we will consider the applied charge current as parameter of interest. Implicit Euler time stepping with constant time step size Δt leads to the $N := T/\Delta t$ nonlinear equation systems

$$\begin{bmatrix} \frac{1}{\Delta t}(c_\mu^{(n+1)} - c_\mu^{(n)}) \\ 0 \end{bmatrix} + A_\mu\left(\begin{bmatrix} c_\mu^{(n+1)} \\ \phi_\mu^{(n+1)} \end{bmatrix}\right) = 0, \qquad (c_\mu^{(n)}, \phi_\mu^{(n)}) \in V_h \oplus V_h. \tag{13.2}$$

The equation systems are solved using a standard Newton iteration scheme.

Note that we can decompose A_μ as

$$A_\mu = A_\mu^{(aff)} + A^{(bv)} + A^{(1/c)} \tag{13.3}$$

where $A^{(bv)}, A^{(1/c)} : V_h \oplus V_h \to V_h \oplus V_h$ accumulate the numerical fluxes corresponding to (13.1) and $\kappa \frac{1-t_+}{F} RT \frac{1}{c} \nabla c$. Thus, the operator $A_\mu^{(aff)}$ collecting the remaining numerical fluxes is affine linear and the only operator in the decomposition depending on the charge rate. $A_\mu^{(aff)}$ can be further decomposed as

$$A_\mu^{aff} = A^{const} + \mu \cdot A^{bnd} + A^{lin}, \tag{13.4}$$

with constant, non-parametric operators A^{const}, A^{bnd} corresponding to the boundary conditions and a non-parametric linear operator $A^{(lin)}$.

13.4 Reduced Basis Approximation

As reduced model we consider the Galerkin projection of (13.2) onto an appropriate reduced basis space $\tilde{V} \subseteq V_h \oplus V_h$, i.e. we solve

$$P_{\tilde{V}} \left\{ \begin{bmatrix} \frac{1}{\Delta t}(\tilde{c}_\mu^{(n+1)} - \tilde{c}_\mu^{(n)}) \\ 0 \end{bmatrix} + A_\mu \left(\begin{bmatrix} \tilde{c}_\mu^{(n+1)} \\ \tilde{\phi}_\mu^{(n+1)} \end{bmatrix} \right) \right\} = 0, \quad (\tilde{c}_\mu^{(n)}, \tilde{\phi}_\mu^{(n)}) \in \tilde{V}, \tag{13.5}$$

where $P_{\tilde{V}}$ denotes the orthogonal projection onto $\tilde{V}$. In order to obtain at an online efficient scheme, the projected operator $P_{\tilde{V}} \circ A_\mu$ has to be approximated by an efficiently computable approximation. Considering the decompositions (13.3) and (13.4), only the nonlinear operators $A_\mu^{(bv)}$, $A_\mu^{(1/c)}$ require special treatment for which we employ empirical operator interpolation [8] based on the empirical interpolation method [2]. Denoting the discrete time differential operator by B, the fully reduced scheme is then given as

$$\begin{aligned} \Big\{ & P_{\tilde{V}} \circ B + P_{\tilde{V}} \circ A^{(const)} + \mu \cdot P_{\tilde{V}} \circ A^{(bnd)} + P_{\tilde{V}} \circ A^{(lin)} \\ & + \{P_{\tilde{V}} \circ I_{M^{(1/c)}}^{(1/c)}\} \circ \tilde{A}_{M^{(1/c)}}^{(1/c)} \circ R_{M'^{(1/c)}}^{(1/c)} \\ & + \{P_{\tilde{V}} \circ I_{M^{(bv)}}^{(bv)}\} \circ \tilde{A}_{M^{(bv)}}^{(bv)} \circ R_{M'^{(bv)}}^{(bv)} \quad \Big\} \left(\begin{bmatrix} \tilde{c}_\mu^{(t+1)} \\ \tilde{\phi}_\mu^{(t+1)} \end{bmatrix} \right) = 0, \end{aligned} \tag{13.6}$$

where $\tilde{A}_{M^{(*)}}^{(*)} : \mathbb{R}^{M'^{(*)}} \to \mathbb{R}^{M^{(*)}}$ $(* = bv, 1/c)$ denotes the restriction of $A^{(*)}$ to certain $M^{(*)}$ image degrees of freedom given the required $M'^{(*)}$ source degrees of freedom, $R_{M'^{(*)}}^{(*)} : V_h \oplus V_h \to \mathbb{R}^{M'^{(*)}}$ is the linear operator restricting finite volume functions to these $M'^{(*)}$ source degrees of freedom, and $I_{M^{(*)}}^{(*)} : \mathbb{R}^{M^{(*)}} \to V_h \oplus V_h$ is the linear interpolation operator to the $M^{(*)}$ evaluation points and an appropriately selected interpolation basis. Note that for the considered finite volume scheme we have $M'^{(*)} \leq 14 \cdot M^{(*)}$, thus $\tilde{A}_{M^{(*)}}^{(*)}$ can be computed quickly for sufficiently small $M^{(*)}$.

The remaining terms in (13.6) are linear (or constant) and can be pre-computed for a given reduced basis of $\tilde{V}$.

In [15] we have considered the solution of (13.6) where $\tilde{V}$ and the interpolation data for $A^{(bv)}$, $A^{(1/c)}$ have been generated using standard model order reduction techniques. The reduced space $\tilde{V}$ was determined by computing a proper orthogonal decomposition (POD) [17] of solution trajectories of (13.2) for an equidistant training set of charge rate parameters. Since the c and ϕ variables are defined on different scales, the POD had to be applied separately for both variables, yielding a reduced space of the form $\tilde{V} = \tilde{V}_c \oplus \tilde{V}_\phi$, in order to obtain a stable scheme. Moreover, the intermediate stages of the Newton algorithms used for solving (13.2) were included in the snapshot data to ensure the convergence of the Newton algorithms when solving the reduced scheme.

The interpolation bases and interpolation points have been obtained by evaluating $A^{(*)}$ on the computed solution trajectories and then performing the EI-GREEDY algorithm [7] on these evaluations. Note that for solution trajectories of (13.2), A_μ vanishes identically in the ϕ-component. Thus, applying the EI-GREEDY algorithm directly to evaluations of A_μ would not have yielded usable interpolation spaces.

13.5 Localized Basis Generation

Localized RB methods can be seen as RB schemes where the reduced space $\tilde{V}$ has a certain direct sum decomposition $\tilde{V} = \tilde{V}_1 \oplus \ldots \oplus \tilde{V}_K$ with subspaces $\tilde{V}_i$ associated to some partition $\overline{\Omega} = \overline{\Omega_1} \cup \ldots \cup \overline{\Omega_K}$ of the computational domain Ω. Since this imposes an additional constraint on the possible choices of reduced spaces $\tilde{V}$, it is not to be expected that such methods yield better approximation spaces for the same (total) dimension of $\tilde{V}$ than classical RB methods. However, these methods can yield enormous saving in computation time during basis generation. In particular, we are interested in the following aspects:

1. When the parametrization of the problem mainly affects the global solution behaviour, only few global solution snapshots may be required to observe all relevant local behaviour. This can be exploited by computing local approximation spaces from global solutions which have been decomposed according to the partition $\Omega_1 \cup \ldots \cup \Omega_K$ (e.g. [1]).
2. The local approximation spaces $\tilde{V}_i$ may be enriched by solving appropriate local problems on a neighbourhood of Ω_i. The solution of the local problems can be trivially parallelized, and each local problem will be solvable much faster than the global problem, which might even be unsolvable with the available computational resources (e.g. [14]).
3. When the problem undergoes local changes (e.g. geometry change due to Li-plating), the spaces $\tilde{V}_i$ which are unaffected by the change can be reused and only few new local problems have to be solved (e.g. [5]).

For many applications, the time for basis generation must be taken into account when considering the overall efficiency of the reduction scheme. Hence, such localization approaches can be an essential tool for making model order reduction profitable for these applications. This is also the case for battery simulations, where typically only relatively few parameter samples are required to gain an appropriate idea of the behaviour of the model and these same computational resources are available for all required simulations. Also note that while reduced system matrices/Jacobians are dense matrices for standard RB approaches, one typically obtains block sparse matrices for localized RB approaches, so the increased global system dimension can be largely compensated by appropriate choices of linear solvers.

In this contribution we investigate if spatially resolved electrochemistry simulations are in principle amenable to such localization techniques. For this we partition the computational domain with a cuboid macro grid with elements $\Omega_1, \dots \Omega_K$ that are aligned with the microscale voxel grid of the given finite volume discretization (cf. Fig. 13.4). This partition induces a direct sum decomposition of V_h:

$$V_h = V_{h,1} \oplus \dots \oplus V_{h,K}, \qquad V_{h_i} = \{f \in V_h \mid \operatorname{supp}(f) \subseteq \overline{\Omega_i}\}.$$

We now compute local reduced spaces $\tilde{V}_{c,i}$, $\tilde{V}_{\phi,i}$ by first computing global solution snapshots $c_{\mu_s}^{(n)}$, $\phi_{\mu_s}^{(n)}$ for preselected parameters $\mu_1, \dots, \mu_S$ and then performing local PODs of the L^2-orthogonal projections of these snapshots onto the local finite volume spaces $V_{h,i}$. Hence,

$$\tilde{V}_{c,i} \subseteq \operatorname{span}\{P_{V_{h,i}}(c_{\mu_s}^{(n)}) \mid 1 \leq s \leq S,\ 1 \leq n \leq N\},$$

$$\tilde{V}_{\phi,i} \subseteq \operatorname{span}\{P_{V_{h,i}}(\phi_{\mu_s}^{(n)}) \mid 1 \leq s \leq S,\ 1 \leq n \leq N\}.$$

Since our high-dimensional model is already given as a non-conforming discretization, we con now simply obtain a reduced model by solving (13.5) with the reduced space

$$\tilde{V} = (\tilde{V}_{c,1} \oplus \dots \oplus \tilde{V}_{c,K}) \oplus (\tilde{V}_{\phi,1} \oplus \dots \oplus \tilde{V}_{\phi,K}).$$

In order to obtain a fully localized model, localized treatment of the nonlinearities $A^{(bv)}$, $A^{(1/c)}$ is required as well. Not only will most of the speedup during the offline phase be lost when the interpolation data is computed without localization. Global interpolation basis vectors will also induce a coupling between all local approximation spaces $\tilde{V}_{c,i}$, $\tilde{V}_{\phi,i}$. Thus the block sparsity structure of the Jacobians appearing in the Newton update problems for solving (13.6) is lost, strongly deteriorating reduced solution times. Moreover, the additional reduced degrees of freedom due to localization can exhibit a destabilizing effect when not accounted for while generating the interpolation spaces: in the limit when each subdomain Ω_i corresponds to a single voxel, we have $\tilde{V} = V_h \oplus V_h$ whereas the images of the interpolated operators are only $M^{(bv)}/M^{(1/c)}$-dimensional with $M^{(bv)}/M^{(1/c)} \ll \dim(V_h \oplus V_h)$.

As a first approach to localized treatment of the nonlinear operators, we proceed similar to the reduced basis generation. We first construct local empirically interpolated operators $I^{(*)}_{i,M_i^{(*)}} \circ \tilde{A}^{(*)}_{i,M_i^{(*)}} \circ R^{(*)}_{i,M_i'^{(*)}}$ $(* = bv, 1/c)$ by applying the EI-GREEDY algorithm to the projected evaluations

$$\{P_{V_{h,i}}(A^{(*)}([c^{(n)}_{\mu_s}, \phi^{(n)}_{\mu_s}]^T)) \mid 1 \leq s \leq S,\ 1 \leq n \leq N\}.$$

We then obtain the localized interpolated operators

$$A^{(*)} \approx I^{(*)} \circ \tilde{A}^{(*)} \circ R^{(*)},$$

where

$$I^{(*)} = \left[I^{(*)}_{1,M_1^{(*)}}, \ldots, I^{(*)}_{K,M_K^{(*)}}\right], \quad \tilde{A}^{(*)} = \operatorname{diag}\left(\tilde{A}^{(*)}_{1,M_1^{(*)}}, \ldots, \tilde{A}^{(*)}_{K,M_K^{(*)}}\right), \tag{13.7}$$

$$R^{(*)} = \left[R^{(*)}_{1,M_1'^{(*)}}, \ldots, R^{(*)}_{K,M_K'^{(*)}}\right]^T. \tag{13.8}$$

Using these operators in (13.6) leads to a basic, fully localized and fully reduced approximation scheme for (13.2).

In order to obtain a stable reduced scheme, accurate approximation of the Butler-Volmer fluxes is crucial. However, each localized interpolated operator only takes interface fluxes into its associated domain Ω_i into account: Let T_1 be a finite volume cell at the boundary of Ω_i and T_2 an adjacent cell in a different subdomain Ω_j, $i \neq j$. Unless both cells are selected as interpolation points for the respective operators, local mass conservation will be violated at the T_1/T_2 interface due to the errors introduced by empirical interpolation.

To investigate whether these jumps in the interface fluxes of the interpolated operators have a destabilizing effect, we consider the following modified scheme: We denote by $A_i'^{(*)} : V_h \oplus V_h \to V_h \oplus V_h$, $(* = bv, 1/c)$ the operator which accumulates all numerical fluxes associated with $A^{(*)}$ which correspond to grid faces contained in $\overline{\Omega_i}$. Fluxes corresponding to faces which are also contained in some $\overline{\Omega_j}$, $i \neq j$, are scaled by $1/2$. This scaling ensures that we have

$$A^{(*)} = \sum_{i=1}^{K} A_i'^{(*)}.$$

Each operator $A_i'^{(*)}$ is interpolated separately yielding approximations $I'^{(*)}_{i,M_i^{(*)}} \circ \tilde{A}'^{(*)}_{i,M_i^{(*)}} \circ R'^{(*)}_{i,M_i'^{(*)}}$, where the interpolation data is again obtained via EI-GREEDY algorithms for the evaluations

$$\left\{A'^{(*)}([c_{\mu_s}^{(n)}, \phi_{\mu_s}^{(n)}]^T) \mid 1 \leq s \leq S,\ 1 \leq n \leq N\right\}.$$

We then proceed as before by defining $I'^{(*)}$, $\tilde{A}'^{(*)}$, $R'^{(*)}$ as in (13.7), and (13.8), obtaining the localized approximation $A^{(*)} \approx I'^{(*)} \circ \tilde{A}'^{(*)} \circ R'^{(*)}$.

13.6 Numerical Experiment

As a first numerical experiment we consider the small battery geometry depicted in Fig. 13.2. For this geometry we compare the performance of the two localized RB approximation schemes introduced in Sect. 13.5 with the scheme without localization described in Sect. 13.4.

The model was simulated for 2000 s with equidistant time steps of size $\Delta t := 10\,s$. In order to preclude any effects from possibly insufficient sampling of the solution manifold, the reduced models were constructed using the relatively large amount of $S = 20$ equidistant parameters in the parameter domain $\mathcal{P} := [0.00012, 0.0012]\,A/cm^2 \approx [0.1, 1]\,C$. All reduced approximation and interpolation spaces were computed with relative POD/EI-GREEDY error tolerances of 10^{-7}. The resulting local reduced basis dimensions for the concentration and potential variables are depicted in Fig. 13.4. The maximum model reduction errors were estimated by computing the reduction errors for a test set of 10 random parameters and are shown for the concentration variable in Fig. 13.3 (the errors in the potential variable show similar behaviour). All simulations of the high-dimensional finite volume battery model have been performed within the DUNE software framework [3, 4], which has been integrated with our model order reduction library pyMOR [11].

We observe (Fig. 13.3, top row) that both localized schemes yield stable reduced order models with good error decay, provided a sufficiently large number of interpolation points is chosen. The localized scheme with special treatment of the

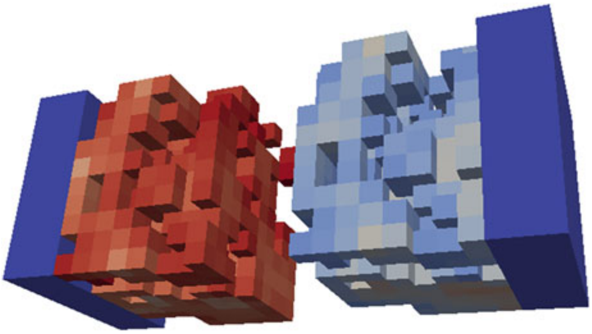


Fig. 13.2 Small battery geometry used in numerical experiment. Domain: $104\mu m \times 40\mu m \times 40\mu m$, 4.600 degrees of freedom. Coloring: Li$^+$-concentration at final simulation time $T = 2000s$, electrolyte not depicted

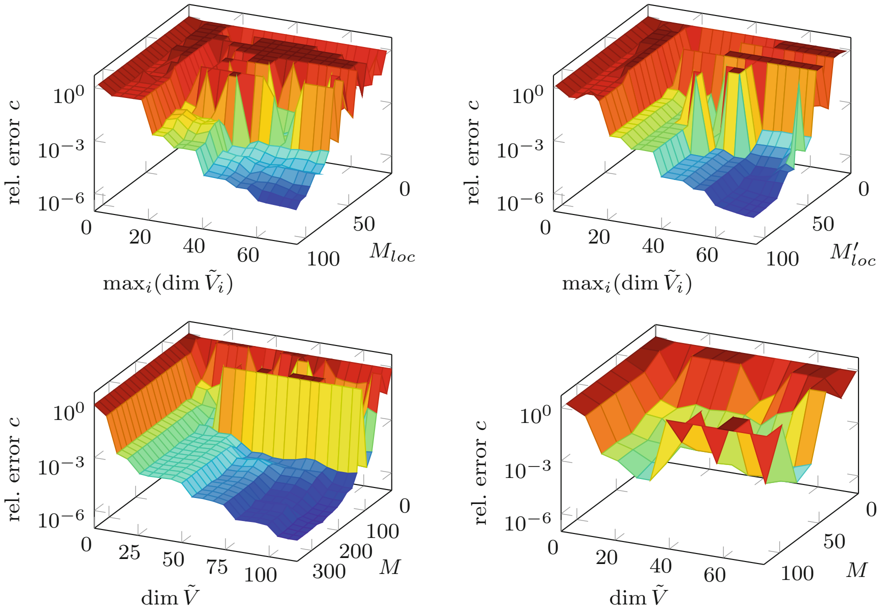


Fig. 13.3 Relative model order reduction errors for the concentration variable c. The error is measured in the L^2-in space, L^∞-in time, L^∞-in μ norm for 10 randomly sampled parameters $\mu \in \mathcal{P} := [0.00012, 0.0012]\, A/cm^2 \approx [0.1, 1]\, C$. *Top left*: errors for the fully localized scheme, $\tilde{V}_i := \tilde{V}_{c,i} \oplus \tilde{V}_{\phi,i}, M_{loc} := \max_i(\max(M_i^{(bv)}, M_i^{(1/c)}))$. *Top right*: errors for the fully localized scheme with additional special treatment of the interface fluxes, $M'_{loc} := \max_i(\max(M_i'^{(bv)}, M_i'^{(1/c)}))$. *Bottom left*: errors for reduced basis approximation without localization, $M := \max(M^{(bv)}, M^{(1/c)})$. *Bottom right*: errors for reduced basis approximation without localization with same axis scaling as in top row

boundary fluxes (top right) is indeed overall more stable than the localized scheme without boundary treatment (top left) and yields slightly smaller reduction errors.

In comparison to the global RB approximation (bottom right), less reduced basis vectors/interpolation points are required per subdomain to obtain a good approximation for the localized schemes. As expected for localized RB schemes, the total number of basis vectors/interpolation points is larger (cf. bottom left) than for the global scheme, however. Given the small size of the full order model, we cannot expect any speedup for the localized reduced models. Nevertheless, based on our experience with global RB approximation of this model [15], we expect only a small increase in the number of required basis vectors/interpolation points to approximate larger, finely resolved geometries. Thus, good speedups can be expected for large-scale applications. Verifying this hypothesis, as well as developing algorithms for efficient localized construction and enrichment of the local approximation spaces, will be subject to future work (Fig. 13.4).

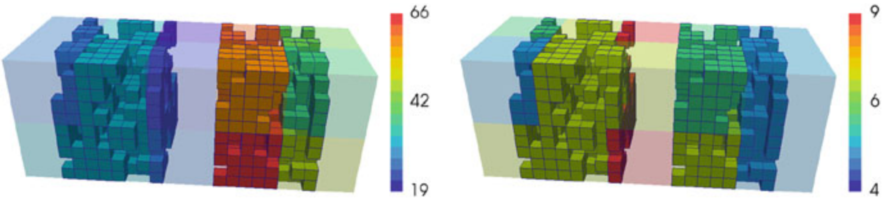


Fig. 13.4 Maximum local reduced basis dimensions $\dim(\widetilde{V}_{c,i})$ (*left*) and $\dim(\widetilde{V}_{\phi,i})$ (*right*) obtained in the numerical experiment

13.7 Conclusion

In this contribution we demonstrated the applicability of the Localized Reduced Basis Method for an instationary nonlinear finite volume Li-ion battery model with resolved pore scale electrode geometry. To this end, we have extended the Localized Reduced Basis Method to parabolic systems of equations, while simultaneously employing the localized empirical operator interpolation in order to deal with the strong nonlinearities of the underlying electrochemical reaction processes. Numerical experiments were given to demonstrate the model order reduction potential of this approach.

Acknowledgements This work has been supported by the German Federal Ministry of Education and Research (BMBF) under contract number 05M13PMA.

References

1. Albrecht, F., Haasdonk, B., Ohlberger, M., Kaulmann, S.: The localized reduced basis multiscale method. In: Proceedings of Algoritmy 2012, Conference on Scientific Computing, Vysoke Tatry, Podbanske, September 9–14, 2012. pp. 393–403 (2012)
2. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. C. R. Math. Acad. Sci. Paris **339**(9), 667–672 (2004)
3. Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Kornhuber, R., Ohlberger, M., Sander, O.: A generic grid interface for parallel and adaptive scientific computing. II. Implementation and tests in DUNE. Computing **82**(2–3), 121–138 (2008)
4. Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Ohlberger, M., Sander, O.: A generic grid interface for parallel and adaptive scientific computing. I. Abstract framework. Computing **82**(2–3), 103–119 (2008)
5. Buhr, A., Engwer, C., Ohlberger, M., Rave, S.: ArbiLoMod: local solution spaces by random training in electrodynamics. In: Benner, P., et al. (eds.) Model Reduction of Parametrized Systems. MS&A, vol. 17. Springer International Publishing, Cham (2017)
6. Buhr, A., Engwer, C., Ohlberger, M., Rave, S.: ArbiLoMod, a simulation technique designed for arbitrary local modifications. SIAM J. Sci. Comput. (2017). Accepted for publication. arXiv e-prints (1512.07840). http://arxiv.org/abs/1512.07840

7. Drohmann, M., Haasdonk, B., Ohlberger, M.: Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. SIAM J. Sci. Comput. **34**(2), A937–A969 (2012)
8. Haasdonk, B., Ohlberger, M., Rozza, G.: A reduced basis method for evolution schemes with parameter-dependent explicit operators. Electron. Trans. Numer. Anal. **32**, 145–161 (2008)
9. Hein, S., Latz, A.: Influence of local lithium metal deposition in 3d microstructures on local and global behavior of lithium-ion batteries. Electrochim. Acta **201**, 354–365 (2016)
10. Latz, A., Zausch, J.: Thermodynamic consistent transport theory of li-ion batteries. J. Power Sources **196**(6), 3296–3302 (2011)
11. Milk, R., Rave, S., Schindler, F.: pyMOR - generic algorithms and interfaces for model order reduction. SIAM J. Sci. Comput. **38**(5), S194–S216 (2016)
12. Ohlberger, M., Rave, S., Schmidt, S., Zhang, S.: A model reduction framework for efficient simulation of li-ion batteries. In: Fuhrmann, J., Ohlberger, M., Rohde, C. (eds.) Finite Volumes for Complex Applications VII-Elliptic, Parabolic and Hyperbolic Problems. Springer Proceedings in Mathematics & Statistics, vol. 78, pp. 695–702. Springer, Switzerland (2014)
13. Ohlberger, M., Rave, S., Schindler, F.: True error control for the localized reduced basis method for parabolic problems. In: Benner, P., et al. (eds.) Model Reduction of Parametrized Systems. MS&A, vol. 17. Springer International Publishing, Cham (2017)
14. Ohlberger, M., Schindler, F.: Error control for the localized reduced basis multiscale method with adaptive on-line enrichment. SIAM J. Sci. Comput. **37**(6), A2865–A2895 (2015)
15. Ohlberger, M., Rave, S., Schindler, F.: Model reduction for multiscale lithium-ion battery simulation. In: Bülent, K., Murat, M., Münevver, T.-S., Serdar, G., Ömür, U. (eds.) Numerical Mathematics and Advanced Applications - ENUMATH 2015, pp. 317–331. Springer International Publishing, Cham (2016). doi:10.1007/978-3-319-39929-4_31, ISBN:978-3-319-39929-4. http://dx.doi.org/10.1007/978-3-319-39929-4_31
16. Popov, P., Vutov, Y., Margenov, S., Iliev, O.: Finite volume discretization of equations describing nonlinear diffusion in li-ion batteries. In: Dimov, I., Dimova, S., Kolkovska, N. (eds.) Numerical Methods and Applications. Lecture Notes in Computer Science, vol. 6046, pp. 338–346. Springer, Berlin/Heidelberg (2011)
17. Sirovich, L.: Turbulence and the dynamics of coherent structures. i-coherent structures. ii-symmetries and transformations. iii-dynamics and scaling. Q. Appl. Math. **45**(3), 561–571 (1987)

Chapter 14
A-Posteriori Error Estimation of Discrete POD Models for PDE-Constrained Optimal Control

Martin Gubisch, Ira Neitzel, and Stefan Volkwein

Abstract In this work a-posteriori error estimates for linear-quadratic optimal control problems governed by parabolic equations are considered. Different error estimation techniques for finite element discretizations and model-order reduction are combined to validate suboptimal control solutions from low-order models which are constructed by a Galerkin discretization and the application of proper orthogonal decomposition. The theoretical findings are used to design an updating algorithm for the reduced-order models; the efficiency and accuracy are illustrated by numerical tests.

14.1 Introduction

Many optimal control problems governed by partial differential equations (PDEs), especially those in higher dimensions, are challenging to be solved numerically because their discretization leads to very high-dimensional problems. This is the reason why model reduction techniques, such as the method of proper orthogonal decomposition (POD), are subject to active research. The method of POD approximates a high-dimensional problem by a smaller, tractable problem by means of projections of the dynamical system onto subspaces that inherit characteristics of the expected solution. Regarding convergence results for POD solutions to parabolic PDEs we refer, for instance, to [14]. In [9], an overview over the topic of POD model order reduction is provided.

Recently, some effort has been made to derive a-priori and a-posteriori error analysis for the reduced control problems. We refer to [11] for a-priori error estimates for POD approximations to control problems, and to [25] for first results

M. Gubisch (✉) • S. Volkwein
Fachbereich Mathematik und Statistik, Universität Konstanz, Universitätsstraße 10, 78457 Konstanz, Germany
e-mail: martin.gubisch@uni-konstanz.de; stefan.volkwein@uni-konstanz.de

I. Neitzel
Institut für Numerische Simulation, Rheinische Friedrich-Wilhelms-Universität Bonn, Wegelerstr. 6, 53115 Bonn, Germany
e-mail: neitzel@ins.uni-bonn.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_14

on a-posteriori error estimates for linear-quadratic problems. These results are extended to nonconvex problems in [13], and to problems with mixed control-state constraints in [8]. The numerical analysis of POD a-posteriori error estimation for optimal control problems is investigated in [23]. However, the available results in the literature on POD a-posteriori error estimates so far do not account for the fact that the subspace for the reduced model is generated from snapshots of the full order model, which, in the setting of PDE constrained optimization, is typically a finite element (FE) discretization of a continuous model problem. Thus, what is commonly referred to as the *true solution* in the context of reduced order models (ROM) is itself in fact an approximation of the *real solution*.

For the FE approximation of the real solution of parabolic problems, there exists a range of results on a-priori and a-posteriori error estimates. Concerning a-priori estimates, we refer to [18] and references therein, where error estimates are provided for a space-time FE discretization of PDE-constrained optimal control problems without further control or state constraints. This approach has been extended to problems with additional control constraints in [19], that—after minor modifications to the simplified structure of the control space—covers the model problem to be discussed in our paper. The discretization therein includes a (discontinuous Galerkin type) variant of the implicit Euler scheme for the time discretization and usual H^1-conforming FE discretization in space. We will heavily rely on this Galerkin structure of the discretization in Sect. 14.4, where we derive our main result. We note in passing that FE discretization error estimates for parabolic problems are available also for certain types of state constraints, see [21] or [7], or semilinear parabolic problems with pointwise-in-time state constraints, cf. [15], or control constraints, see e.g. [22]. Cf. also results on plain convergence without any rates in [2].

However, standard a-priori estimates are in general not a good indicator for enlarging or updating the reduced-order models since they usually tend to overestimate the effective errors significantly. Instead, a-posteriori error estimates may be applied; [1] provides such methods for uncontrolled parabolic PDEs, focussing on adaptive time and spatial grid selection. In [12], a detailed discussion about the discretization and regularization of optimal control problems is presented. For adaptive discretization strategies based on a-posteriori error estimates for parabolic optimal control without additional inequality constraints we refer to e.g. [17] and [20] and the references therein.

In this paper, we are concerned with the following model problem with state y and control u in spaces to be specified,

$$\text{Minimize } J(y,u) := \frac{1}{2}\int_0^T \int_\Omega (y(t,x) - y_d(t,x))^2 dx\, dt + \frac{\nu}{2}\sum_{i=1}^{N_u}\int_0^T u_i(t)^2\, \mathrm{d}t \tag{14.1a}$$

subject to

$$\partial_t y - \Delta y = \sum_{i=1}^{N_u} u_i \chi_i + f \text{ in } (0,T) \times \Omega, \qquad y(0,\cdot) = y_0 \text{ in } \Omega, \tag{14.1b}$$

$$y = 0 \text{ in } (0,T) \times \partial\Omega, \qquad u_a \leq u(t) \leq u_b \text{ almost everywhere in } (0,T) \tag{14.1c}$$

where the last inequality is to be understood componentwise. Note that the functions $\chi_i : \Omega \to \mathbb{R}$, $i = 1, \ldots, N_u$, are given, fixed data. The exact setting of the model problem will be described in the next section. We are interested in discussing how the discretization error on the one hand and the model order reduction error on the other hand relate and can possibly be balanced, motivated by the following two questions:

1. Since a reduced-order model is based on the snapshots of a high-dimensional approximative PDE solution, the model already includes the FE discretization errors and it does not make sense to decrease the POD residual below the order of the FE discretization error.
2. If, however, the error of the POD approximation does not reflect the order of the discretization error even when increasing the POD basis rank, i.e. the size of the reduced order model, the current POD basis may not reflect the dynamics of the optimal POD basis $\bar{\psi}$ referring to the optimal state solution $\bar{y}$. In this case, an update of the POD basis may be required to improve the results.

The paper is organized as follows: In Sect. 14.2, we summarize available theoretical results for the continuous optimal control problem. Then, in Sect. 14.3, we briefly describe the FE discretization along the lines of [19], and state an a-priori error estimate. In short, the discretization is utilized by discretizing the state and adjoint equations by the so-called dG(0)cG(1) method, cf. for instance [4, 5]. That is, the time-discretization of the PDEs is done by piecewise constant functions, whereas usual H^1-conforming finite elements in space are used. The time-dependent controls are discretized implicitly via the optimality conditions, cf. also the variational discretization approach in [10]. For convenience of the reader, we summarize existence and regularity results for the solution of the semidiscrete and fully discrete state equations, as well as existence of unique solutions to the optimal control problems and optimality conditions on each discretization level. For our main result in the subsequent section, we will use in particular the optimality conditions from Sect. 14.3.2 as well as the error estimate from [19], which we state in Sect. 14.3.3. Our main result, the a-posteriori error analysis for a suboptimal discrete solution, where we have in mind the discrete POD solution, follows in Sect. 14.4. The main step in our analysis is to extend the a-posteriori error analysis technique from [25], which is related to the one in [16] for ordinary differential equations, to estimate the error between the full-order FE solution and the (discrete) solution of the reduced model. Here, we readily make use of the fact that the dG(0)cG(1) discretization is a Galerkin scheme. We eventually obtain an a-posteriori error estimator along the lines of [25], but in contrast to the latter paper, the estimator is computable in the strict sense since it only depends on the computed, FE solution rather than the true unknown continuous solution. Comparing this error to the FE error can be an indication if for instance a POD basis update is useful. Then, Sect. 14.5 finally describes the method of model order reduction via proper orthogonal decomposition. We end this paper by numerical experiments in Sect. 14.6.

It seems reasonable to include the discussion of a-posteriori finite element error analysis in future work. While the focus of our paper is on a computable a-posteriori error estimate for the solution of the reduced-order model, the clear separation of discretization and model-order reduction errors will allow to balance the latter with any type of available a-posteriori discretization error estimate.

14.2 Optimization Problem

In the following, we lay out the principle assumptions on the data in (14.1) and summarize known results on existence and regularity of solutions to the underlying PDE, the control problem itself, as well as first-order necessary and, due to convexity, also sufficient optimality conditions.

Assumption 2.1 *Let $\Omega \subset \mathbb{R}^n$, $n = 1, 2, 3$, be a convex polygonal or polyhedral domain with boundary $\partial\Omega$ for $n = 2, 3$, or an open interval in $\mathbb{R}$. Moreover, let $T > 0$ be a given real number that defines the time interval $I := (0, T)$. In addition, $\nu \in \mathbb{R}$ is a positive, fixed parameter, and the bounds $u_a, u_b \in \mathbb{R}^{N_u}$ are vectors of real numbers that fulfill $u_a < u_b$ componentwise. The desired state y_d is a function from $L^2(I \times \Omega)$ and the initial state y_0 is a function from $H_0^1(\Omega)$. For the shape functions $\chi_i \colon \Omega \to \mathbb{R}$, $i = 1, \dots, N_u$, we require $\chi_i \in H_0^1(\Omega)$.*
We introduce the following short notation for inner products and norms on the spaces $L^2(\Omega)$ and $L^2(I \times \Omega)$, as well as $L^2(I; \mathbb{R}^{N_u})$:

$$\begin{aligned} (v, w) &:= (v, w)_{L^2(\Omega)}, & (v, w)_I &:= (v, w)_{L^2(I\times\Omega)}, & \langle v, w\rangle_I &:= \langle v, w\rangle_{L^2(I;\mathbb{R}^{N_u})} \\ \|v\| &:= \|v\|_{L^2(\Omega)}, & \|v\|_I &:= \|v\|_{L^2(I\times\Omega)}, & |v|_I &:= \|v\|_{L^2(I;\mathbb{R}^{N_u})}. \end{aligned}$$

Throughout the paper we abbreviate $V := H_0^1(\Omega)$; $c > 0$ will denote generic auxiliary constants. Moreover, in order to find a weak formulation of the state equation (14.1b) and the optimal control problem (14.1), we introduce the state space Y, the control space U, and the set of admissible controls U_{ad},

$$Y := W(0, T) = \{v \mid v \in L^2(I, V) \text{ and } \partial_t v \in L^2(I, V^*)\}, \qquad U := L^2(I, \mathbb{R}^{N_u}),$$
$$U_{\mathrm{ad}} := \{u \in U \mid u_a \le u(t) \le u_b \text{ for a.a. } t \in I \text{ componentwise}\},$$

as well as the control operator

$$B\colon U \to L^2(I \times \Omega), \quad u \mapsto \sum_{i=1}^{N_u} u_i(t)\chi_i(x).$$

A weak formulation of the state equation (14.1b) for a fixed control $u \in U$ and fixed initial state $y_0 \in V$ as well as source term $f \in L^2(I \times \Omega)$ is to find a state $y \in Y$ that satisfies

$$\int_0^T (\partial_t y, \varphi)_{V^*,V}\, \mathrm{d}t + (\nabla y, \nabla \varphi)_I = (Bu, \varphi)_I + (f, \varphi)_I \; \forall \varphi \in L^2(I, V), \; y(0, \cdot) = y_0. \tag{14.2}$$

The following existence and regularity result is readily available from [6].

Proposition 1 *For fixed control $u \in U$, fixed source term $f \in L^2(I \times \Omega)$, and fixed initial state $y_0 \in V$ there exists a unique solution $y \in Y$ of the weak state equation* (14.2)*. Moreover, the solution exhibits the improved regularity*

$$y \in L^2(I, H^2(\Omega) \cap V) \cap H^1(I, L^2(\Omega)) \hookrightarrow C(\bar{I}, V)$$

and the stability estimate

$$\|\partial_t y\|_I + \|y\|_I + \|\nabla y\|_I + \|\nabla^2 y\|_I \le C(|u|_I + \|f\|_I + \|\nabla y_0\|) \tag{14.3}$$

is satisfied for a constant $C > 0$.

By the regularity $y \in H^1(0, T; L^2(\Omega))$ it is justified to use the bilinear form

$$\mathbf{b}(y, \varphi) := (\partial_t y, \varphi)_I + (\nabla y, \nabla \varphi)_I,$$

and use the weak formulation

$$\mathbf{b}(y, \varphi) = (Bu, \varphi)_I + (f, \varphi)_I \quad \forall \varphi \in Y, \qquad y(0, \cdot) = y_0. \tag{14.4}$$

Note that due to the linearity of the state equation, (14.1) can be reformulated equivalently into a setting with homogeneous initial condition and without additional source term f by splitting the solution of (14.4) into two parts $y = \hat{y} + y_u$, which fulfill the PDEs

$$\partial_t \hat{y} - \Delta \hat{y} = f \text{ in } I \times \Omega, \quad \hat{y}(0, \cdot) = y_0 \text{ in } \Omega, \quad \hat{y} = 0 \text{ in } I \times \partial\Omega, \tag{14.5}$$

as well as

$$\partial_t y_u - \Delta y_u = \sum_{i=1}^{N_u} u_i(t)\chi_i(x) \text{ in } I \times \Omega, \quad y_u(0,\cdot) = 0 \text{ in } \Omega, \quad y_u = 0 \text{ in } I \times \partial\Omega \tag{14.6}$$

in the weak sense. The fixed term $\hat{y}$ is independent of the controls and can be incorporated into the desired state y_d. For ease of presentation, we will therefore assume without loss of generality: $y_0 = 0$ as well as $f = 0$. In the following we will, however, state more general results from [18, 19] for nonhomogeneous initial conditions and additional source terms.

Next we introduce the linear control-to-state mapping $S : U \to Y$, $Su = y_u$, which leads to the reduced objective function $\hat{J} : U \to \mathbb{R}_0^+$ with $u \mapsto J(S(u), u)$. Note that here and in the following, we tacitly use the operator S also if we interpret the state y as a function in $L^2(I \times \Omega)$. This makes (14.1) equivalent to

$$\text{Minimize } \hat{J}(u) \text{ subject to } u \in U_{\text{ad}}. \tag{\textbf{P}}$$

The following existence and uniqueness result is obtained by standard arguments, cf. for instance [24], since the set of admissible controls is not empty by Ass. 2.1.

Lemma 1 *Let Assumption 2.1 be satisfied. Then the optimal control problem* (**P**) *admits a unique optimal control $\bar{u} \in U_{\text{ad}}$ with associated optimal state $\bar{y} = S\bar{u}$.*
Let us refer to [24] for a detailed proof. We proceed by discussing standard first order necessary optimality conditions for the optimal control problem with the help of a variational inequality. Due to convexity, these conditions are also sufficient for optimality.

Lemma 2 *A control $\bar{u}$ is the unique solution of* (**P**) *if and only if $\bar{u} \in U_{\text{ad}}$ and the following variational inequality holds:*

$$\hat{J}'(\bar{u})(u - \bar{u}) \geq 0 \quad \forall u \in U_{\text{ad}}. \tag{14.7}$$

For a proof, we refer again to e. g., [24]. In order to express the optimality conditions in a more convenient way we define for any control $u \in U_{\text{ad}}$ the adjoint state variable $p = p(u) \in Y$, which is the solution of

$$-\partial_t p - \Delta p = y_d - y(u) \text{ in } I \times \Omega, \quad p(T,\cdot) = 0 \text{ in } \Omega, \quad p = 0 \text{ in } I \times \partial\Omega \tag{14.8}$$

with $y(u) = Su$. A weak solution of this adjoint problem can be defined by means of the already introduced bilinear form **b**, since elements of Y can be integrated by parts in time. We obtain

$$\mathbf{b}(\varphi, p) = (\varphi, y_d - y(u))_I \quad \forall \varphi \in Y, \qquad p(T,\cdot) = 0. \tag{14.9}$$

Note that Prop. 1 is applicable to (14.9) after a time transformation $\tau = T - t$. We then rewrite the first-order optimality conditions from Prop. 2 in the form

$$\langle \nu\bar{u} - B^* p(\bar{u}), u - \bar{u}\rangle_I \geq 0 \qquad \forall u \in U_{\text{ad}},$$

where $B^*: L^2(I \times \Omega) \to U$ denotes the Hilbert space adjoint operator of B satisfying the formula $(Bu, v)_I = \langle u, B^* v\rangle_I$ for all $u, v \in U$. The following identity for B^* can easily be verified by means of the above definition. For any $\varphi \in L^2(I \times \Omega)$, we find

$$B^*\varphi = u \in L^2(I, \mathbb{R}^{N_u}), \quad u_i(t) := \int\limits_{\Omega} \varphi(t,x)\chi_i(x)dx, \quad i = 1, \ldots, N_u \text{ and } t \in I.$$

Then, using the pointwise projection on the admissible set,

$$P_{\text{ad}}: U \to U_{\text{ad}}, \quad P_{\text{ad}}(u)_i(t) := \max(u_a, \min(u_b, u_i(t))), \quad i = 1, \ldots, N_u,$$

the optimality condition simplifies further to

$$\bar{u} = P_{\text{ad}}\left(\frac{1}{\nu}B^* p(\bar{u})\right). \tag{14.10}$$

We refer to [24] for the technique of proof. From Prop. 1 and the projection formula, we deduce the following regularity result, which is a direct consequence of Prop. 2.3 in [19] taking into account that the controls depend only on the time variable.

Proposition 2 *Let $\bar{u}$ be the solution of the optimization problem (**P**) with associated state $\bar{y} = y(\bar{u})$ and let $\bar{p} := p(\bar{u})$ denote the corresponding adjoint state. Then $\bar{u}, \bar{y}, \bar{p}$ achieve the following regularities:*

$$\bar{y}, \bar{p} \in L^2(I, H^2(\Omega) \cap V) \cap H^1(I, L^2(\Omega)) \hookrightarrow C(\bar{I}, V), \; \bar{u} \in H^1(I; \mathbb{R}^{N_u}) \hookrightarrow C(\bar{I}; \mathbb{R}^{N_u})$$

14.3 Discretization of the Problem

This section is devoted to the FE discretization of the optimal control problem under consideration. We review the discretization procedure as well as results on e.g. stability of discrete solutions, and an a-priori error estimate for the controls on the continuous and discrete level from e.g. [19] for linear-quadratic problems with controls varying in space and time, or [22], where the state equation is nonlinear, but the setting of only time-dependent controls is addressed explicitly. More precisely, we first give a brief overview about semidiscretization of the state (and adjoint) equation in time by piecewise constant functions, with values in V. Note that the resulting scheme is a variant of the implicit Euler method. Even though the control functions themselves will not be discretized explicitly, we will obtain, by means of the semidiscrete optimality conditions, that the optimal control is in fact piecewise constant in time, and therefore a discrete function. In a second step, we discretize the involved PDEs also in space. Here, we use usual H^1-conforming linear finite elements. The resulting discretization scheme is commonly referred to as dG(0)cG(1) method. Since the control functions are functions in time, only, solving the time-and-space discrete optimality system corresponds to solving a completely discretized problem.

14.3.1 Semidiscretization in Time

Along the lines of [18, 19], let a partitioning of the time interval $\bar{I} = [0, T]$ be given as $\bar{I} = \{0\} \cup I_1 \cup I_2 \cup \ldots \cup I_M$ with subintervals $I_m = (t_{m-1}, t_m]$ of size k_m, defined by time points $0 = t_0 < t_1 < \ldots < t_{M-1} < t_M = T$. The discretization parameter k is defined as a piecewise constant function by setting $k_{|I_m} = k_m$ for $m = 1, 2, \ldots, M$, yet k also denotes the maximal size of the time steps, i.e., $k = \max k_m$. The semidiscrete trial and test space is given by

$$Y_k = \{v_k \in L^2(I, V) \mid v_{k|I_m} \in \mathscr{P}_0(I_m, V),\ m = 1, 2, \ldots, M\},$$

where $\mathscr{P}_0(I_m, V)$ denotes the space of constant functions defined on I_m with values in V. The control space U and the set of admissible controls U_{ad} remain unchanged, yet we will later find that the semidiscrete optimal control is an element of the space

$$U_k := \{v_k \in U \mid v_{k|I_m} \in \mathscr{P}_0(I_m, \mathbb{R}^{N_u}),\ m = 1, 2, \ldots, M\}.$$

For functions $v_k \in Y_k$ we define

$$v_{k,m}^+ := \lim_{t \to 0+} v_k(t_m + t) = v_k(t_{m+1}), \quad v_{k,m}^- := \lim_{t \to 0+} v_k(t_m - t) = v_k(t_m),$$

$$[v_k]_m := v_{k,m}^+ - v_{k,m}^- = v_k(t_{m+1}) - v_k(t_m) =: v_{k,m+1} - v_{k,m},$$

and introduce the short notation $(v, w)_{I_m} := (v, w)_{L^2(I_m \times \Omega)}$, $\|v\|_{I_m} := \|v\|_{L^2(I_m \times \Omega)}$ for functions $v, w \in L^2(I_m \times \Omega)$. The semidiscrete version of the bilinear form $\mathbf{b}(\cdot, \cdot)$ for $y_k, \varphi_k \in Y_k$ is given by

$$\mathbf{b}_k(y_k, \varphi_k) = (\nabla y_k, \nabla \varphi_k)_I + \sum_{m=2}^{M} (y_{k,m} - y_{k,m-1}, \varphi_m) + (y_{k,1}, \varphi_{k,1}),$$

and the dG(0) semidiscretization of the state equation (14.4) for fixed control $u \in U$ reads as follows: Find a state $y_k = y_k(u) \in Y_k$ such that

$$\mathbf{b}_k(y_k, \varphi_k) = (Bu, \varphi_k)_I + (f, \varphi_k)_I + (y_0, \varphi_{k,1}) \quad \forall \varphi_k \in Y_k. \tag{14.11}$$

Here, $\varphi_{k,m}$ denotes $\varphi_{k_{|I_m}}$. Note that we assumed $y_0 = 0$ and $f = 0$ for ease of presentation, but include the more general setting in the following theorem.

Theorem 1 *For every fixed control $u \in U$, the semidiscrete state equation* (14.11) *with potentially nonhomogenous initial state $y_0 \in V$ and source term $f \in L^2(I \times \Omega)$ admits a unique semidiscrete solution $y_k \in Y_k$ satisfying the stability result*

$$\|y_k\|_I^2 + \|\nabla y_k\|_I^2 + \|\Delta y_k\|_I^2 + \sum_{i=1}^{M} k_m^{-1} \|y_{k,i} - y_{k,i-1}\|^2 \le C\{|u|_I^2 + \|f\|_I^2 + \|\nabla y_0\|^2\}$$

with a constant $C > 0$ independent of the discretization parameters.

Proof This is a direct consequence of Theorem 4.1 in [18], taking into account that the controls are only time-dependent functions, independent of the spatial variable. With the semidiscrete control-to-state operator $S_k: U \to Y_k$, $S_k(u) = y_k$, where y_k is the solution of (14.11), and consequently a semidiscrete reduced objective function $\hat{J}_k : U \to \mathbb{R}_0^+, u \mapsto J(S_k(u), u)$, the reduced semidiscrete problem formulation reads

$$\text{Minimize } \hat{J}_k(u) \text{ subject to } u \in U_{\text{ad}}. \qquad (\mathbf{P}_k)$$

Existence of a unique semidiscrete optimal control $\bar{u}_k \in U_{\text{ad}}$ with associated semidiscrete optimal state $\bar{y}_k \in Y_k$, analogously to Problem (**P**), follows by standard arguments. Likewise, we obtain first-order necessary and sufficient optimality conditions for $\bar{u}_k \in U_{\text{ad}}$ in the form

$$\hat{J}_k'(\bar{u}_k)(u - \bar{u}_k) = \langle \nu \bar{u}_k - B^* p_k(\bar{u}_k), u - \bar{u}_k \rangle_I \ge 0 \quad \forall u \in U_{\text{ad}}, \qquad (14.12)$$

where $p_k = p_k(u) \in Y_k$ is the semidiscrete adjoint state, i.e. the solution of the semidiscrete adjoint equation

$$\mathbf{b}_k(\varphi_k, p_k) = (\varphi_k, y_d - y_k(u))_I \qquad \forall \varphi_k \in Y_k. \qquad (14.13)$$

Note that the stability results from Theorem 1 are applicable to (14.13). By making use of the projection formula

$$\bar{u}_k = P_{\text{ad}}\left(\frac{1}{\nu} B^* p_k(\bar{u}_k)\right) \qquad (14.14)$$

on this level of discretization we readily obtain a structural result for the semidiscrete optimal control:

Corollary 1 *From the projection formula* (14.14) *we deduce that all components of the optimal control $\bar{u}_k$ are piecewise constant in time, i.e. $\bar{u}_k \in U_k \cap U_{\text{ad}}$.*
Note that from Corollary 1, it is clear that $\bar{u}_k$ also solves the problem

$$\text{Minimize } \hat{J}_k(u_k) \text{ subject to } u_k \in U_k \cap U_{\text{ad}},$$

and thus the time discretization of the controls does not have to be discussed explicitly.

14.3.2 Discretization in Space

Now, we introduce the spatial discretization of the optimal control problem, still in the spirit of e.g. [19]. We consider two- or three-dimensional shape regular meshes; see, e.g., [3], consisting of quadrilateral or hexahedral cells K, which constitute a nonoverlapping cover of the computational domain Ω. For $n = 1$, the cells reduce to subintervalls of Ω. We denote the mesh by $\mathscr{T}_h = \{K\}$ and define the discretization parameter h as a cellwise constant function by setting $h_{|_K} = h_K$ with the diameter h_K of the cell K. We use the symbol h also for the maximal cell size, i.e., $h = \max h_K$. On the mesh $\mathscr{T}_h$ we construct a conforming FE space $V_h \subset V$ in the standard way

$$V_h = \{v \in V \mid v_{|_K} \in \mathscr{Q}(K) \text{ for } K \in \mathscr{T}_h\},$$

with basis $\{\Phi_h^j\}_{j=1,\dots,N}$, where $\mathscr{Q}(K)$ consists of shape functions obtained via bilinear transformations of polynomials up to degree one defined on a reference cell $\hat{K}$; cf. also Sect. 3.2 in [18]. Then, the space-time discrete finite element space

$$Y_{kh} = \{v_{kh} \in L^2(I, V_h) \mid v_{kh|_{I_m}} \in \mathscr{P}_0(I_m, V_h),\ m = 1, 2, \dots, M\} \subset Y_k.$$

leads to a discrete version of the bilinear form $\mathbf{b}_k(\cdot,\cdot)$ for $y_{kh}, \varphi_{kh} \in Y_{kh}$, given by

$$\mathbf{b}_{kh}(y_{kh}, \varphi_{kh}) = (\nabla y_{kh}, \nabla \varphi_{kh})_I + \sum_{m=2}^{M} (y_{kh,m} - y_{kh,m-1}, \varphi_{kh,m}) + (y_{kh,1}, \varphi_{kh,1}).$$

Eventually, we obtain the so-called dG(0)cG(1) discretization of the state equation for given control $u \in U$: Find a state $y_{kh} = y_{kh}(u) \in Y_{kh}$ such that

$$\mathbf{b}_{kh}(y_{kh}, \varphi_{kh}) = (Bu, \varphi_{kh})_I + (f, \varphi_{kh})_I + (y_0, \varphi_{kh,1}) \qquad \forall \varphi_{kh} \in Y_{kh}. \tag{14.15}$$

Theorem 2 *Let Ass. 2.1 be satisfied. Then, for each $u \in U$ and possibly nonhomogeneous initial condition $y_0 \in V$ and source term $f \in L^2(\Omega)$, there exists a unique solution $y_{kh} \in Y_{kh}$ of equation (14.15) satisfying the stability estimate*

$$\begin{aligned} &\|y_{kh}\|_I^2 + \|\nabla y_{kh}\|_I^2 + \|\Delta_h y_{kh}\|_I^2 + \sum_{i=1}^{M} k_m^{-1} \|y_{kh,i} - y_{kh,i-1}\|^2 \\ &\le C\left(|u|_I^2 + \|f\|_I^2 + \|\nabla \Pi_h y_0\|^2\right) \end{aligned}$$

with a constant $C > 0$ independent of the discretization parameters k and h.

Proof Again, this follows from [18], see Theorem 4.6, with the obvious modifications due to the structure of the control space.
Here, $\Delta_h \colon V_h \to V_h$ is defined by $(\Delta_h v_h, \varphi_h) = -(\nabla v_h, \nabla \varphi_h)$ for all $\varphi_h \in V_h$. Repeating the steps from the semidiscrete setting leads to the introduction of the discrete control-to-state operator $S_{kh} \colon U \to Y_{kh},\ y_{kh} = S_{kh}(u)$, the discrete reduced objective function $J_{kh} : U_{\text{ad}} \to \mathbb{R}_0^+, \quad u \mapsto J(u, S_{kh}(u))$, and the discrete problem formulation

$$\text{Minimize } \hat{J}_{kh}(u) \text{ subject to } u \in U_{\text{ad}}, \tag{$\mathbf{P}_{kh}$}$$

which, again by standard arguments, admits a unique optimal solution $\bar{u}_{kh} \in U_{\text{ad}}$. First-order necessary and sufficient optimality conditions for $\bar{u}_{kh} \in U_{\text{ad}}$ are given by

$$\hat{J}'_{kh}(\bar{u}_{kh})(u - \bar{u}_{kh}) = \langle \nu \bar{u}_{kh} - B^* p_{kh}(\bar{u}_{kh}), u - \bar{u}_{kh} \rangle_I \geq 0 \quad \forall u \in U_{\text{ad}}, \tag{14.16}$$

via the discrete adjoint state $p_{kh} \in Y_{kh}$ being the solution of the discrete adjoint equation

$$\mathbf{b}_{kh}(\varphi_{kh}, p_{kh}) = (\varphi_{kh}, y_d - y_{kh})_I \qquad \forall \varphi_{kh} \in Y_{kh}. \tag{14.17}$$

Let us conclude that the structure of (14.16), which is analogous to the continuous problem due to Galerkin discretization, is of central importance to adapt the error estimation techniques from [25], see Sect. 14.4.

14.3.3 A-Priori Error Estimate

Let us end this section by stating an a-priori discretization error estimate between the optimal control $\bar{u}$ of (**P**) and the fully discrete solution $\bar{u}_{kh}$ of ($\mathbf{P}_{kh}$). The following theorem is a direct consequence of Theorem 6.1 in [18], where error estimates are proven for control functions $u \in L^2(I \times \Omega)$. The specific setting with finitely many time-dependent controls is considered for nonconvex problems with semilinear state equations in [22], Prop. 5.4.

Theorem 3 *Let $\bar{u}$ be the optimal control of Problem* (**P**) *and $\bar{u}_{kh}$ be the optimal control of Problem* ($\mathbf{P}_{kh}$)*. Then there exists a constant $C > 0$ independent of k and h, such that the following error estimate is satisfied: $|\bar{u} - \bar{u}_{kh}|_I \leq C(k + h^2)$.*

14.4 A-Posteriori Error Analysis for an Approximate Solution to $\mathbf{P}_{kh}$

If the possibly high-dimensional optimization problem ($\mathbf{P}_{kh}$) is not solved directly, but an approximate solution $\bar{u}^p_{kh} \in U_k \cap U_{\mathrm{ad}}$ is obtained by e.g. a POD Galerkin approximation, see Sect. 14.5, one is interested in estimating the error

$$\varepsilon^p_{kh} := |\bar{u}^p_{kh} - \bar{u}_{kh}|_I,$$

without knowing $\bar{u}_{kh}$. Together with an available error estimate for

$$\varepsilon_{kh} := |\bar{u} - \bar{u}_{kh}|_I,$$

this leads to an estimate for the error between the real optimal solution $\bar{u}$ and the computed solution $\bar{u}^p_{kh}$,

$$\varepsilon := |\bar{u} - \bar{u}^p_{kh}|_I \leq \varepsilon_{kh} + \varepsilon^p_{kh},$$

where the influence of the discretization error on the one hand and the model reduction error on the other hand are clearly separated. We will use Theorem 3 for the first part. We point out that ε_{kh} may also be estimated by other available e.g. a-posteriori error estimators, and now focus on developing an estimate for the second part. Since the discretization has been obtained by a Galerkin method, and optimality conditions from Sect. 14.3 are available, we can apply the arguments and techniques from [25]. Utilizing the notation $y^p_{kh} = S_{kh}(u^p_{kh})$ as well as $p^p_{kh} = p_{kh}(u^p_{kh})$, we obtain the following:

Lemma 3 *Let $u^p_{kh} \in U_k \cap U_{\mathrm{ad}}$ be arbitrary. Define a function $\xi^p_k \in U_k$ componentwise by $(\xi^p_k)_i := \big(\nu u^p_{kh} - B^* p^p_{kh}\big)_i$ for $i = 1, \ldots, N_u$ as well as the index sets of active constraints*

$$\mathscr{I}_i^- := \{1 \leq m \leq M \mid (u^p_{kh})_{i|_{I_m}} = u_{a,i}\}, \quad i = 1, \ldots, N_u,$$
$$\mathscr{I}_i^+ := \{1 \leq m \leq M \mid (u^p_{kh})_{i|_{I_m}} = u_{b,i}\}, \quad i = 1, \ldots, N_u,$$

the active sets $\mathscr{A}_i^- := \bigcup\{I_m \mid m \in \mathscr{I}_i^-\}$, $\mathscr{A}_i^+ := \bigcup\{I_m \mid m \in \mathscr{I}_i^+\}$ and the inactive set $\mathscr{A}_i^\circ := I \setminus (\mathscr{A}_i^- \cup \mathscr{A}_i^+)$. Then the function $\zeta_k \in U_k$ defined componentwise by

$$\zeta_{k,i} = [\xi^p_{k,i}]_- = -\min\{0, \xi^p_{k,i}\} \text{ on } \mathscr{A}_i^-,$$
$$\zeta_{k,i} = -[\xi^p_{k,i}]_+ = -\max\{0, \xi^p_{k,i}\} \text{ on } \mathscr{A}_i^+, \quad \zeta_{k,i} = -\xi^p_{k,i} \text{ on } \mathscr{A}_i^\circ$$

for $i = 1, \ldots, N_u$ satisfies the perturbed variational inequality

$$\langle \nu u^p_{kh} - B^* p^p_{kh} + \zeta_k, u - u^p_{kh} \rangle_I \geq 0 \qquad \forall u \in U_k \cap U_{\mathrm{ad}}.$$

Proof Note first that due to the piecewise constant time discretization, the function ζ_k is an element of U_k. Now, direct calculations for $u \in U_k \cap U_{\mathrm{ad}}$ shows:

$$\langle \nu u^p_{kh} - B^* p^p_{kh} + \zeta_k, u - u^p_{kh} \rangle = \sum_{i=1}^{N_u} \sum_{m=1}^{M} \int_{I_m} (\xi^p_k + \zeta_k)_i (u - u^p_{kh})_i \, dt$$

$$= \sum_{i=1}^{N_u} \sum_{m \in \mathscr{I}_i^-} k_m (\xi^p_{k,m} + [\xi^p_{k,m}]_-)_i (u - u_a)_i + \sum_{m \in \mathscr{I}_i^+} k_m (\xi^p_{k,m} - [\xi^p_{k,m}]_+)_i (u - u_b)_i \geq 0,$$

where we have used that $(u - u_a)_i \geq 0$ and $(u - u_b)_i \leq 0$.

Theorem 4 *Let $\bar{u}_{kh}$ be the optimal solution to* (**P**$_{kh}$) *with associated state $\bar{y}_{kh}$ and adjoint state $\bar{p}_{kh}$. Suppose that $u^p_{kh} \in U_k \cap U_{\mathrm{ad}}$ is chosen arbitrarily with associated state $y^p_{kh} = y_{kh}(u^p_{kh}) \in Y_{kh}$ and adjoint state $p^p_{kh} = p_{kh}(u^p_{kh}) \in Y_{kh}$, and let $\zeta_k \in U_k$ be given as in Lemma 3. Then the following estimate is satisfied: $|\bar{u}_{kh} - u^p_{kh}|_I \leq |\zeta_k|_I / \nu$.*

Proof The variational inequality (14.15)–(14.17) and Lemma 3 imply

$$\begin{aligned}
0 &\leq \langle \nu \bar{u}_{kh} - B^* \bar{p}_{kh}, u^p_{kh} - \bar{u}_{kh} \rangle_I + \nu \langle u^p_{kh} - B^* p^p_{kh} + \zeta_k, \bar{u}_{kh} - u^p_{kh} \rangle_I \\
&= -\nu |\bar{u}_{kh} - u^p_{kh}|_I^2 + \langle B^*(p^p_{kh} - \bar{p}_{kh}), u^p_{kh} - \bar{u}_{kh} \rangle_I - \langle \zeta_k, u^p_{kh} - \bar{u}_{kh} \rangle_I \\
&= -\nu |\bar{u}_{kh} - u^p_{kh}|_I^2 + (B(u^p_{kh} - \bar{u}_{kh}), p^p_{kh} - \bar{p}_{kh})_I - \langle \zeta_k, u^p_{kh} - \bar{u}_{kh} \rangle_I \\
&= -\nu |\bar{u}_{kh} - u^p_{kh}|_I^2 + \mathbf{b}(y^p_{kh} - \bar{y}_{kh}, p^p_{kh} - \bar{p}_{kh}) - \langle \zeta_k, u^p_{kh} - \bar{u}_{kh} \rangle_I \\
&= -\nu |\bar{u}_{kh} - u^p_{kh}|_I^2 - \|y^p_{kh} - \bar{y}_{kh}\|_I^2 - \langle \zeta_k, u^p_{kh} - \bar{u}_{kh} \rangle_I.
\end{aligned}$$

From this calculation, we conclude that $\nu |\bar{u}_{kh} - u^p_{kh}|_I^2 \leq |\zeta_k|_I |u^p_{kh} - \bar{u}_{kh}|_I$ yields $|\bar{u}_{kh} - u^p_{kh}|_I \leq |\zeta_k|_I / \nu$.

Combining the results of Theorem 3 and Theorem 4, we directly obtain

Corollary 2 *Let $\bar{u} \in U_{\mathrm{ad}}$ be the optimal control of Problem* (**P**), *let $u^p_{kh} \in U_k \cap U_{\mathrm{ad}}$ be chosen arbitrarily, and let $\zeta_k \in U_k$ be given as in Lemma 3. Then there exists a constant $C > 0$ independent of k and h, such that the following error estimate is fulfilled: $|\bar{u} - u^p_{kh}|_I \leq C(k + h^2) + |\zeta_k|_I / \nu$.*

14.5 The POD Galerkin Discretization

In this section, we construct a problem specific subspace $V^\ell_h \subseteq V_h$ with significantly smaller dimension $\dim V^\ell_h = \ell \ll N = \dim V_h$ such that the projection of an element y_{kh} on the reduced state space $Y^\ell_{kh} = \{\phi_{kh} \in Y_{kh} \mid \forall m = 1, \ldots, M : \phi_{kh}|_{I_m} \in \mathscr{P}_0(I_m, V^\ell_h)\}$ is still a good approximation of y_{kh}. More precisely, for a given basis rank ℓ, we choose orthonormal ansatz functions $\psi^1_h, \ldots, \psi^\ell_h \in V_h$

such that $y_{kh} - \mathscr{P}_h^\ell y_{kh}$ is minimized with respect to $\|\cdot\|_{L^2(I,V_h)}$ where $\mathscr{P}_h^\ell : V_h \to V_h^\ell = \operatorname{span}(\psi_h^1, \dots, \psi_h^\ell)$ denotes the canonical projector $\mathscr{P}_h^\ell(y_{kh}(t)) = \sum_{l=1}^{\ell} \langle y_{kh}(t), \psi_h^l \rangle_{V_h} \psi_h^l$. Hence, these basis functions $\psi_h^1, \dots, \psi_h^\ell$ are given as a solution to the optimization problem

$$\min_{\psi_h^1,\dots,\psi_h^\ell \in V_h} \int_0^T \|y_{kh}(t) - \mathscr{P}_h^\ell y_{kh}(t)\|_{V_h}^2 \mathrm{d}t \qquad \text{subject to} \qquad \langle \psi_h^i, \psi_h^j \rangle_{V_h} = \delta_{ij}, \tag{14.18}$$

where δ_{ij} denotes the Kronecker delta. Since the integrand is piecewise constant on the time intervals $I_1, \dots, I_m$, we replace (14.18) by

$$\min_{\psi_h^1,\dots,\psi_h^\ell \in V_h} \sum_{m=1}^{M} k_m \|y_{kh,m} - \mathscr{P}_h^\ell y_{kh,m}\|_{V_h}^2 \qquad \text{subject to} \qquad \langle \psi_h^i, \psi_h^j \rangle_{V_h} = \delta_{ij}. \tag{14.19}$$

A solution to problem (14.18) is called a *rank-ℓ POD basis* to the trajectory $y_{kh} \in Y_{kh}$ and can be determined by solving the corresponding eigenvalue problem

$$\mathscr{R}(y_{kh})\psi_h = \lambda \psi_h, \qquad \mathscr{R}(y_{kh}) = \int_0^T \langle \cdot, y_{kh}(t) \rangle_{V_h} y_{kh}(t)\, \mathrm{d}t, \tag{14.20}$$

choosing $\psi_h^1, \dots, \psi_h^\ell \in V_h$ as the eigenfunctions of $\mathscr{R}_{kh} := \mathscr{R}(y_{kh}) : V_h \to V_h$ corresponding to the ℓ largest eigenvalues $\lambda_1 \geq \cdots \geq \lambda_\ell > 0$, cf. [8], Sect. 2.2. Notice that the adjoint operator $\mathscr{R}_{kh}^*$ is compact due to the Hilbert-Schmidt theorem, i.e. $\mathscr{R}_{kh}$ is compact as well, and that $\mathscr{R}_{kh}$ is non-negative, so a complete decomposition of V_h into eigenfunctions of $\mathscr{R}_{kh}$ is available, and each eigenvalue except for possibly zero has finite multiplicity, see [8], Lemma 2.12 and Theorem 2.13.

The following approximation property holds, cf. Proposition 3.3 in [14]:

$$\int_0^T \|y_{kh}(t) - \mathscr{P}_h^\ell y_{kh}(t)\|_{V_h}^2 \,\mathrm{d}t = \sum_{m=1}^{M} \left\| y_{kh,m} - \mathscr{P}_h^\ell y_{kh,m} \right\|_{V_h}^2 = \sum_{l=\ell+1}^{\operatorname{rank}(\mathscr{R}_{kh})} \lambda^l.$$

After a POD basis of some reference trajectory $y_{kh} \in Y_{kh}$ is constructed, we introduce the *reduced state space*

$$Y_{kh}^\ell = \{\phi_{kh} \in Y_{kh} \mid \forall m = 1, \dots, M : \phi_{kh}|_{I_m} \in \mathscr{P}_0(I_m, V_h^\ell)\}$$

and consider the reduced-order optimization problem

$$\min_{(y_{kh}^\ell, u) \in Y_{kh}^\ell \times U_{\mathrm{ad}}} \frac{1}{2} \sum_{m=1}^{M} k_m \|y_{kh,m} - y_{d,m}\|_{V_h}^2 + \frac{\nu}{2} |u|_I^2. \tag{14.21}$$

The first-order optimality conditions of (14.21), consisting of a reduced state equation corresponding to (14.4), a reduced adjoint state equation corresponding to (14.9) and a control equation corresponding to (14.10), read as

$$\mathbf{b}_{kh}(y_{kh},\varphi_{kh}) - (Bu_{kh},\varphi_{kh})_I = 0 \qquad \forall \varphi_{kh} \in Y_{kh}^\ell, \tag{14.22a}$$

$$\mathbf{b}_{kh}(\varphi_{kh},p_{kh}) - (\varphi_{kh},y_d - y_{kh})_I = 0 \qquad \forall \varphi_{kh} \in Y_{kh}^\ell, \tag{14.22b}$$

$$\langle \nu u_{kh} - B^* p_{kh}, u - u_{kh}\rangle_I \geq 0 \qquad \forall u \in U_{\mathrm{ad}}. \tag{14.22c}$$

Since the *optimal* trajectory $\bar{y}_{kh}$ is not available in practice to build up an appropriate POD basis for the reduced-order model, different methods have been developed recently on how to construct a reference trajectory $\tilde{y}_{kh}$ which covers enough dynamics of $\bar{y}_{kh}$ to build up an accurate reduced order space V_h^ℓ.

If the desired state y_d is smooth and the regularization parameter ν is sufficiently large, the dynamics of $\bar{y}_{kh}$ are usually simple enough such that the state solution $\tilde{y}_{kh}$ to some more or less arbitrary reference control such as $\tilde{u} \equiv 1$ generates a suitable reduced space V_h^ℓ. Otherwise, if $\tilde{y}_{kh}$ differs too much from the optimal solution $\bar{y}_{kh}$, it may be necessary to choose an inefficiently large basis rank ℓ to represent the more complex dynamics of $\bar{y}_{kh}$ in the eigenfunctions of $\tilde{y}_{kh}$. Moreover, though the properties of $\mathscr{R}(\tilde{y}_{kh})$ should guarantee a suitable approximation of $\bar{y}_{kh}$ in V_h^ℓ if ℓ is sufficiently large, numerical instabilities arise especially if λ_ℓ comes close to zero: In this case, the set of eigenfunctions to $\mathscr{R}(\tilde{y}_{kh})$ corresponding to the nonzero eigenvalues is not enlarged by a basis of range$(\mathscr{R})^\perp$; instead, numerical noise is added on the eigenfunctions so that the POD basis is not improved, but even perturbed, cf. Fig. 14.4. Consequently, the system matrices of the reduced-order model become singular and the reduced order solutions get instable. One way to balance out this problem is to provide a *basis update* which improves the reference control $\tilde{y}_{kh}$ and hence the ansatz space V_h^ℓ.

Let $(\psi_h^1,\dots,\psi_h^\ell) \subseteq V_h$ be a rank-ℓ POD basis computed from some reference state $\tilde{y}_{kh}$ and let u_{kh}^ℓ, $\ell = 1,2,\dots$, denote the control solution to the reduced-order optimality system (14.22). We study the development of the control errors $\varepsilon_{\mathrm{ex}}^\ell = |\bar{u} - \bar{u}_{kh}^\ell|_I$ for various ℓ. A stagnation of (the order of) $\varepsilon_{\mathrm{ex}}^\ell$ may be caused by three different effects:

1. The chosen basis ranks are still too small to represent the corresponding optimal state solutions $\bar{y}_{kh}$ in an appropriate way. Adding some more basis vectors may finally lead to a decay of $\varepsilon_{\mathrm{ex}}^\ell$; the stagnation is not necessarily an indication for a badly chosen reference trajectory. Indeed, even the *optimal* POD basis corresponding to the exact FE solution $\bar{y}_{kh}$ does not guarantee small errors $\varepsilon_{\mathrm{ex}}^\ell$ for small basis ranks, cf. our numerical tests in the next section.
2. The vector space spanned by the eigenfunctions of $\mathscr{R}(y_{kh})$ is exploited before $\varepsilon_{\mathrm{ex}}^\ell$ decays below the desired exactness ε. As mentioned above, additional POD elements will not improve the error decay in this case. Further, the available information may not be sufficient to extend the current basis by additional vectors at all.

3. The accuracy of the FE model $\varepsilon_{kh}^{\text{ex}} = |\bar{u} - \bar{u}_{kh}|_I$ is reached. In this case, expanding the POD basis may decrease the error $\varepsilon_{kh}^{\ell} = |\bar{u}_{kh} - \bar{u}_{kh}^{\ell}|_I$ between the high-dimensional and the low-dimensional approximation of $\bar{u}$, but not the actually relevant error $\varepsilon_{\text{ex}}^{\ell}$ between $\bar{u}_{kh}^{\ell}$ and the exact control solution $\bar{u}$.

14.6 Numerical Results

We test our findings combined in Algorithm 1* with the aid of an analytical test problem where the exact control, state and adjoint solutions $\bar{u}, \bar{y}, \bar{p}$ are known explicitly. For this purpose, we introduce a desired control $u_d \in U$ in addition and replace the objective functional by $\tilde{J}(y, u) = J(y, u - u_d)$. In the optimality system, this has no impact on the state equation or the adjoint equation; the adapted variational inequality for the control now reads as $\langle \nu(\bar{u} - u_d) - B^*\bar{p}, u - \bar{u} \rangle_I \geq 0$ for all $u \in U_{\text{ad}}$. We choose the one-dimensional spatial domain $\Omega = (0, 2\pi)$, the control space $U = L^2(I, \mathbb{R}^1)$ consisting of a single-component control $u = u(t)$ on the time interval $I = [0, \frac{\pi}{2}]$, the single shape function $\chi(x) = \sin(x)$, the lower and upper control bounds $u_a = -5$, $u_b = 5$ and the regularization parameter $\nu = 1$.

Algorithm 1 *

```
Require: Basis ranks $\ell_{\min} < \ell_{\max}$, initial POD basis elements $\psi_h^1, \dots, \psi_h^{\ell_{\max}} \in V_h$, maximal
    number of basis updates $j_{\max}$.
Estimate FE error $\varepsilon_{kh}^{\text{ex}} = |\bar{u} - \bar{u}_{kh}|_I$. Set $j = 1$, $\ell = \ell_{\min}$.
while $j \leq j_{\max}$ do
    Set $\psi_h^{\ell} = (\psi_1, \dots, \psi_{\ell})$.
    Calculate optimal control $\bar{u}_{kh}^{\ell}$ to the $\ell$-dimensional reduced-order model.
    Estimate ROM residual $\varepsilon_{kh}^{\ell} = |\bar{u}_{kh} - \bar{u}_{kh}^{\ell}|_I$.
    if $\varepsilon_{kh}^{\ell} \leq \varepsilon_{kh}^{\text{ex}}$ then
        break                                                              (optimal accuracy reached)
    else if $\ell < \ell_{\max}$ then
        Set $\ell = \ell + 1$.                                                     (enlarge POD basis)
    else
        Calculate new POD basis elements $\psi_h^1, \dots, \psi_h^{\ell_{\max}} \in V_h$ of $\mathscr{R}(\bar{y}_{kh}^{\ell})$.   (update POD basis)
        Set $\ell = \ell_{\min}$ and $j = j + 1$.
    end if
end while
```

We propose to choose the initial POD elements by starting with the admissible constant control $u(t) = \frac{1}{2}(u_b - u_a)$, calculating the corresponding state $y_{kh}(u)$ and selecting the first eigenvectors of $\mathscr{R}(y_{kh})$. Further, we suggest to choose a minimal and a maximal basis rank $\ell_{\min}$, $\ell_{\max}$ at the beginning to increase the reduced model rank ℓ frequently, beginning with $\ell_{\min}$, until $\varepsilon_{\text{ex}}^{\ell}$ decays below the desired exactness ε or $\ell_{\max}$ is reached; in the latter case, a basis update is provided, choosing the lastly determined POD optimal control $\bar{u}_{kh}^{\ell_{\max}}$ to calculate new snapshots $\bar{y}_{kh}^{\ell_{\max}}$ and resetting the model rank on $\ell_{\min}$

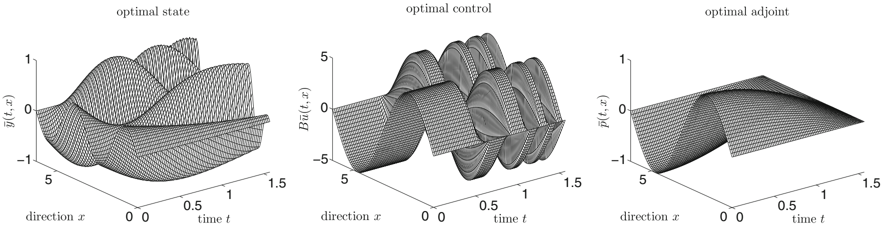


Fig. 14.1 The optimal state $\bar{y}$ (*left*), the optimal control $B\bar{u}$ (*center*) and the optimal adjoint state $\bar{p}$ (*right*) of the test setting

Then the optimal triple (see Fig. 14.1)

$$\bar{y}(t,x) = \sin(x)\cos(x\exp(t)), \qquad \bar{p}(t,x) = \sin(x)\cos(t), \tag{14.23a}$$

$$\bar{u}(t) = P_{\mathrm{ad}}\left(\left\{\tfrac{\pi}{\nu}\cos(t)\right\} + \{10\sin(\exp(2t))\}\right), \tag{14.23b}$$

can be realized by the source term $f \in L^2(I \times \Omega)$, the initial value $y_0 \in H_0^1(\Omega)$, the desired state $y_d \in L^2(I \times \Omega)$ and the desired control $u_d \in L^2(I, \mathbb{R}^1)$ given by

$$\begin{aligned} f(t,x) &= -\sin(x)\sin(xe^t)xe^t + \sin(x)\cos(xe^t) + \cos(x)\sin(xe^t)e^t \\ &\qquad + \cos(x)\sin(xe^t)e^t + \sin(x)\cos(xe^t)e^{2t} - \chi(x)\bar{u}(t), \\ y_0(x) &= \sin(x)\cos(x), \quad y_d(t,x) = \sin(x)\sin(t) + \sin(x)\cos(t) + \bar{y}(t,x), \\ u_d(t) &= 10\sin(\exp(2t)). \end{aligned}$$

By direct recalculation one sees that the functions in (14.23) fulfill the adapted optimality equations. The full-order optimality system (14.15)–(14.17) as well as the reduced-order one (14.22) are solved by the fixpoint iteration $u_{n+1} = F(u_n) = P_{\mathrm{ad}}(B^*p(y(u))/\nu)$ with admissible initial control u_0. This procedure generates a converging sequence $(u_{n+1})_{n\in\mathbb{N}} \subset U_{\mathrm{ad}}$ with limit $\bar{u}$ given that ν is not too small. In this case, F is a contracting selfmapping on U_{ad} and the Banach fixpoint theorem provides decay rates for the residual $|\bar{u} - u_n|_I$, cf. [8, Sect. 5.5]. Compared to numerical strategies which provide higher convergence rates such as Newton methods, the numerical effort within the single iterations is small since no coupled systems of PDEs have to be solved.

14.6.1 Finite Element Error Estimation

In order to be able to combine the a-priori FE error estimates from Sect. 14.3.3 with the a-posteriori error estimate for the POD approximation in a reasonable way, we need to estimate the constant appearing in Theorem 3. More precisely, we will

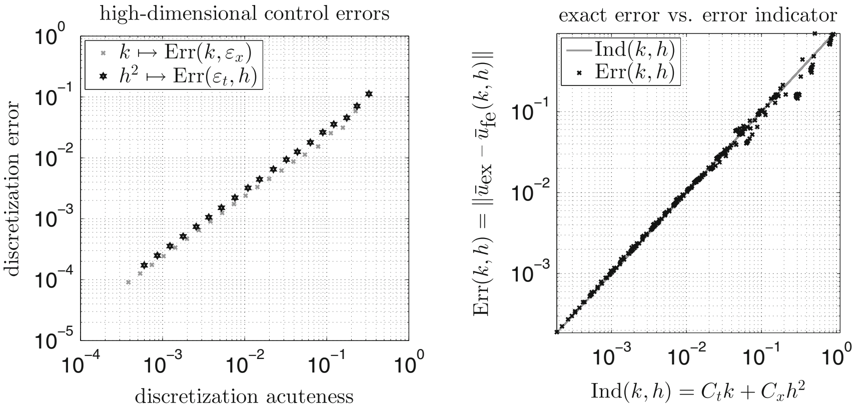


Fig. 14.2 On the *left*, we show that the exact errors of time integration $k \mapsto \varepsilon^{\mathrm{ex}}_{kh_0}$ on a sufficiently fine spatial grid $h_0^2 \ll k$ and the exact spatial errors $h^2 \mapsto \varepsilon^{\mathrm{ex}}_{k_0h}$ with sufficiently small time steps $k_0 \ll h^2$ evolve approximatively linear in logarithmic scales. On the *right*, on sees that $\mathrm{Ind}(k,h) \approx \varepsilon^{\mathrm{ex}}_{kh}$ holds given that the time and space grids are not too coarse: The bounds are sharp, but not rigorous

estimate two constants $C_t, C_x > 0$ such that $\|\bar u - \bar u_{kh}\|_I \approx C_t k + C_x h^2$ holds. In this way, we receive slightly better results then with the constant C presented in Theorem 3; choose $C = \max(C_t, C_x)$ for convenience. The dependency between the time and space discretization quantities h, k and the resulting discretization errors is shown in Fig. 14.2 (left); the quality of this *error indicator* is shown in Fig. 14.2 (right). We estimate such constants C_t, C_x by solving the discretized optimality system (14.15)–(14.17) on grids of different grid widths:

$$C_t = \frac{1}{k_1}|\bar u_{k_1h_0} - \bar u_{k_2h_0}|_I \ (h_0^2, k_2 \ll k_1), \quad C_x = \frac{1}{h_1^2}|\bar u_{k_0h_1} - \bar u_{k_0h_2}|_I \ (h_2^2, k_0 \ll h_1^2).$$

Notice that this procedure does not guarantee that $\mathrm{Ind}(k,h) = C_t k + C_x h^2$ provides an upper bound for the FE error. In our numerical example, we choose the parameters $h_0 = 3.14\text{e-}2$ and $k_1 = 3.08\text{e-}2$, $k_2 = 1.57\text{e-}3$ as well as $k_0 = 3.93\text{e-}3$ and $h_1 = 2.99\text{e-}1$, $h_2 = 6.22\text{e-}2$ and get $C_t = 0.2$, $C_x = 0.184$.

14.6.2 Model Reduction Error Estimation

We divide the time interval into $M = 6400$ subintervals and the spatial domain into $N = 500$ subdomains. In this case, $k = 2.45\text{e-}4$ and $h = 1.26\text{e-}2$ hold. With the growth constants given above, we expect a FE accuracy of the magnitude $\varepsilon^{\mathrm{ex}}_{kh} = 7.82\text{e-}5$. Let $\tilde y_{kh}$ be a perturbation of the optimal FE solution $\bar y_{kh}$ such that $\|\bar y_{kh} - \tilde y_{kh}\|_I$ is of the order $\tilde\varepsilon = 1.00\text{e-}7$. Although $\tilde\varepsilon < \varepsilon^{\mathrm{ex}}_{kh}$ holds true, the POD

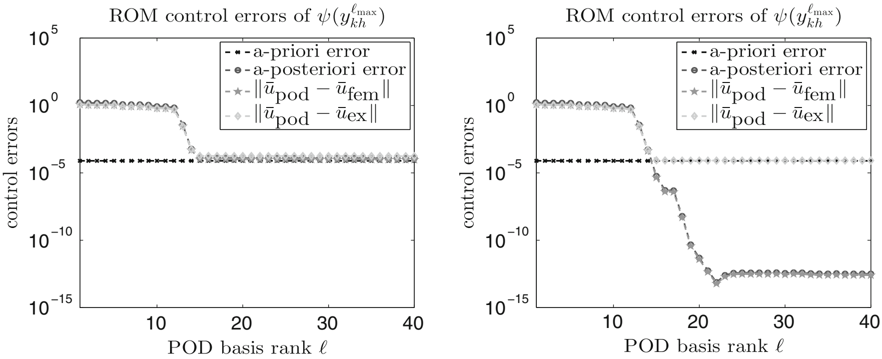


Fig. 14.3 Here we present the decay behavior of the errors ε_{kh}^{ℓ} and $\varepsilon_{\text{ex}}^{\ell}$ in dependence of the chosen POD basis rank ℓ as well as the desired FE error level $\varepsilon_{kh}^{\text{ex}}$. On the *left*, a POD basis belonging to a small pointwise perturbation of the optimal state is applied to build up the reduced order model. On the *right*, we use an updated POD basis

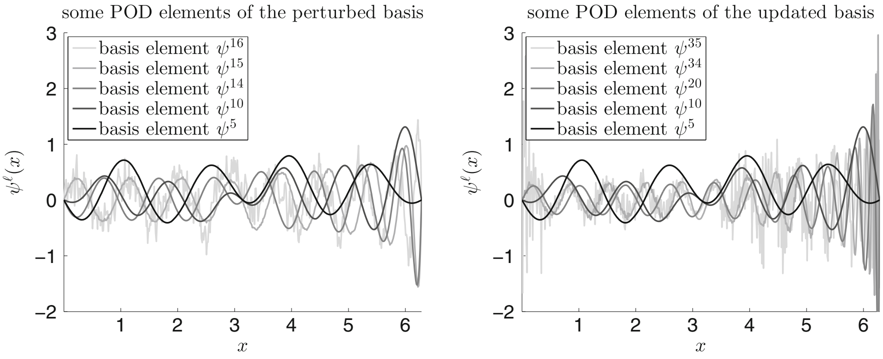
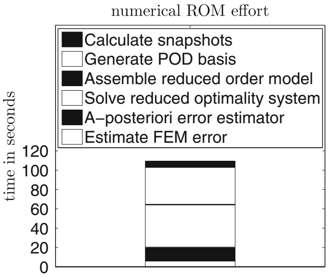


Fig. 14.4 Some elements of the perturbed (*left*) and the updated (*right*) POD basis in comparison

error ε_{kh}^{ℓ} and hence also the exact error $\varepsilon_{\text{ex}}^{\ell}$ between the controls $\bar{u}_{kh}^{\ell}$ and $\bar{u}_{kh}$ or $\bar{u}$, respectively, do not reach the desired accuracy $\varepsilon_{kh}^{\text{ex}}$ independent of the chosen basis rank ℓ, cf. Fig. 14.3 (left); the POD elements react very sensitive if the corresponding snapshots are covered by noise. Notice that a perturbation of the control generating the snapshots would not have this destabilizing effect on the POD basis. After providing a basis update, we observe that the low-order model error ε_{kh}^{ℓ} decays far below the high-order model accuracy $\varepsilon_{kh}^{\text{ex}}$ for increasing basis rank ℓ while the exact error $\varepsilon_{\text{ex}}^{\ell}$ stagnates on the level of $\varepsilon_{kh}^{\text{ex}}$ as expected, cf. Fig. 14.3 (right). Starting with $\ell_{\min} = 12$, the Algorithm stops successfully after three basis extensions without requiring a further basis update. In Fig. 14.4 we compare the two POD bases. It turns out that overall, the first fifteen basis elements coincide, except of possibly the sign. Then, the noise starts to dominate the perturbed basis; the seventeenth basis function has no structure any more (left). In contrast, the updated basis spans a subspace of

Table 14.1 The duration of the single processes within the reduced order modelling routine with adaptive basis selection

Process (ROM)	Time	#	Total
Calculate snapshots	3.17 s	2×	6.34 s
Generate POD basis	19.09 s	2×	38.17 s
Assemble reduced system	0.33 s	2×	0.66 s
Solve reduced system	0.87 s	50×	43.65 s
A-posteriori error estimator	7.28 s	2×	14.56 s
FEM error estimation	6.04 s	1×	6.04 s
Total			109.40 s
Process (FEM)	Time	#	Total
Solve full-order system	25.81 s	30×	774.27 s
Total			774.27 s



V_h with approximative dimension 34. POD elements of higher rank order than 34 get unstable as well (right), but here, the spanned space is already sufficiently large and includes enough dynamics of the optimal state solution to represent $\bar{y}_{kh}$ up to FE precision. Finally, we compare the effort of the full-order model with the reduced one. The calculation times of the single steps are presented in Table 14.1; the calculations are provided on an Intel(R) Core(Tm) i5 2.40 GHz processor. We use $N = 4000$ finite elements now and $M = 4000$ time steps. Without model reduction, 30 fixpoint iterations are required, taking 774.27 s in total. To avoid noise in the POD elements, we choose adaptive basis ranks $\ell_{\min} = \ell_{\max} = \min\{20, \ell_\sigma\}$ where $\ell_\sigma = \max\{\ell \mid \lambda_\ell > 1.0\text{e-}12\}$: No basis elements corresponding to eigenvalues close to zero shall be used to build up the reduced model. As before, the first solving of the reduced optimality system requires 30 iterations, but the model error ε^ℓ_{kh} is still above the desired accuracy $\varepsilon^{\text{ex}}_{kh}$. The rank of the reduced-order problem is 14 (since $\lambda_{15} = 4.03\text{e-}13$). Providing one basis update with the snapshots $y^\ell_{kh}(\bar{u}^{14}_{kh})$, the new POD elements include more dynamics of the problem although the eigenvalues decay as before; now, we have $\ell = 15$ (since $\lambda_{16} = 2.61\text{e-}13$), the fixpoint routine terminates after 20 iterations and ε^ℓ_{kh} decays below the FE accuracy. The reduced-order modelling takes 109.40 s, 14.09% of the full-order solving.

Acknowledgements Ira Neitzel is grateful for the support of her former host institution Technische Universität München. Martin Gubisch gratefully acknowledges support by the German Science Fund DFG grant VO 1658/2-1 *A-posteriori-POD Error Estimators for Nonlinear Optimal Control Problems governed by Partial Differential Equations.*

References

1. Bergam, A., Bernardi, C., Mghazli, Z.: A posteriori analysis of the finite element discretization of some parabolic equations. Math. Comput. **74**(251), 1117–1138 (2004)
2. Chrysafinos, K.: Convergence of discontinuous Galerkin approximations of an optimal control problem associated to semilinear parabolic PDEs. ESAIM: M2AN **44**(1), 189–206 (2010)
3. Ciarlet, P.G.: The Finite Element Method for Elliptic Problems. North-Holland, Amsterdam (1987)
4. Eriksson, K., Johnson, C., Thomée, V.: Time discretization of parabolic problems by the discontinuous Galerkin method. ESAIM: M2AN **19**, 611–643 (1985)
5. Eriksson, K., Estep, D., Hansbo, P., Johnson, C.: Computational Differential Equations. Cambridge University Press, UK (1996)
6. Evans, L.C.: Partial Differential Equations. American Mathematical Society, Providence, RI (1998)
7. Gong, W., Hinze, M.: Error estimates for parabolic optimal control problems with control and state constraints. Comput. Optim. Appl. **56**(1), 131–151 (2013)
8. Gubisch, M., Volkwein, S.: POD a-posteriori error analysis for optimal control problems with mixed control-state constraints. Comput. Optim. Appl. **58**(3), 619–644 (2014)
9. Gubisch, M., Volkwein, S.: Proper orthogonal decomposition for linear-quadratic optimal control. In: Benner, P., Cohen, A., Ohlberger, M., Willcox, K. (eds.) Model Reduction and Approximation: Theory and Algorithms. SIAM, Philadelphia, PA (2017). Preprint available: http://nbn-resolving.de/urn:nbn:de:bsz:352-250378
10. Hinze, M.: A variational discretization concept in control constrained optimization: the linear-quadratic case. Comput. Optim. Appl. **30**(1), 45–61 (2005)
11. Hinze, M., Volkwein, S.: Error estimates for abstract linear-quadratic optimal control problems using proper orthogonal decomposition. C omput. Optim. Appl. **39**(3), 319–345 (2007)
12. Hinze, M., Rösch, A.: Discretization of Optimal Control Problems. In Leugering, G., Engell, S., Griewank, A., Hinze, M., Rannacher, R., Schulz, V., Ulbrich, M., Ulbrich, S. (eds.) Constrained Optimization and Optimal Control for Partial Differential Equations. International Series of Numerical Mathematics, vol. 160, pp. 391–430. Springer, Berlin/Heidelberg (2012)
13. Kammann, E., Tröltzsch, F., Volkwein, S.: A posteriori error estimation for semilinear parabolic optimal control problems with application to model reduction by POD. ESAIM: M2AN **47**(2), 555–581 (2013)
14. Kunisch, K., Volkwein, S.: Galerkin proper orthogonal decomposition methods for parabolic problems. Numer. Math. **90**(1), 117–148 (2001)
15. Ludovici, F., Neitzel, I., Wollner, W.: A priori error estimates for state constrained semilinear parabolic optimal control problems. Submitted. INS Preprint No. 1605. Available at: http://neitzel.ins.uni-bonn.de/pub/LNW2016_INS.pdf (2016)
16. Malanowski, K., Büskens, C., Maurer, H.: Convergence of approximations to nonlinear optimal control problems. In Fiacco, A.V. (ed.) Mathematical Programming with Data Perturbations. Pure and Applied Mathematics, vol. 195, pp. 253–284. CRC Press, Boca Raton, FL (1998)
17. Meidner, D., Vexler, B.: Adaptive space-time finite element methods for parabolic optimization problems. SIAM J. Control. Optim. **46**(1), 116–142 (2007)
18. Meidner, D., Vexler, B.: A priori error estimates for space-time finite element discretization of parabolic optimal control problems part I: problems without control constraints. SIAM J. Control. Optim. **47**(3), 1150–1177 (2008)
19. Meidner, D., Vexler, B.: A priori error estimates for space-time finite element discretization of parabolic optimal control problems part II: problems with control constraints. SIAM J. Control. Optim. **47**(3), 1301–1329 (2008)
20. Meidner, D., Vexler, B.: Adaptive space-time finite element methods for parabolic optimization problems. In Leugering, G., Engell, S., Griewank, A., Hinze, M., Rannacher, R., Schulz, V., Ulbrich, M., Ulbrich, S. (eds.) Constrained Optimization and Optimal Control for Partial Differential Equations. International Series of Numerical Mathematics, vol. 160, pp.319–348. Springer, Berlin/Heidelberg (2012)

21. Meidner, D., Rannacher, R., Vexler, B.: A priori error estimates for space-time finite element discretization of parabolic optimal control problems with pointwise state contstraints in time. SIAM J. Control. Optim. **49**(5), 1961–1997 (2011)
22. Neitzel, I., Vexler, B.: A priori error estimates for space-time finite element discretization of semilinear parabolic optimal control problems. Numer. Math. **120**, 345–386 (2008)
23. Studinger, A., Volkwein, S.: Numerical analysis of POD a-posteriori error estimation for optimal control. In Bredies, K., Clason, C., Kunisch, K., von Winckel, G. (eds.) Control and Optimization with PDE Constraints. International Mathematical Series, vol. 164, pp. 137–158. Springer, Berlin/Heidelberg (2013)
24. Tröltzsch, F.: Optimal Control of Partial Differential Equations: Theory, Methods and Applications. Graduate Studies in Mathematics, vol. 112. American Mathematical Society, Providence, RI (2010)
25. Tröltzsch, F., Volkwein, S.: POD a-posteriori error estimates for linear-quadratic optimal control problems. Comput. Optim. Appl. **44**(1), 83–115 (2009)

Chapter 15
Hi-POD Solution of Parametrized Fluid Dynamics Problems: Preliminary Results

Davide Baroli, Cristina Maria Cova, Simona Perotto, Lorenzo Sala, and Alessandro Veneziani

Abstract Numerical modeling of fluids in pipes or network of pipes (like in the circulatory system) has been recently faced with new methods that exploit the specific nature of the dynamics, so that a one dimensional axial mainstream is enriched by local secondary transverse components (Ern et al., Numerical Mathematics and Advanced Applications, pp 703–710. Springer, Heidelberg, 2008; Perotto et al., Multiscale Model Simul 8(4):1102–1127, 2010; Perotto and Veneziani, J Sci Comput 60(3):505–536, 2014). These methods—under the name of Hierarchical Model (Hi-Mod) reduction—construct a solution as a finite element axial discretization, completed by a spectral approximation of the transverse dynamics. It has been demonstrated that Hi-Mod reduction significantly accelerates the computations without compromising the accuracy. In view of variational data assimilation pro-

D. Baroli
MOX, Dipartimento di Matematica, Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133 Milano, Italy

Faculté des Sciences, de la Technologie et de la Communication, Université du Luxembourg, 6, rue Richard Coudenhove-Kalergi, L-1359, Luxembourg
e-mail: davide.baroli@polimi.it

S. Perotto (✉)
MOX, Dipartimento di Matematica, Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133 Milano, Italy
e-mail: simona.perotto@polimi.it

C.M. Cova
Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133 Milano, Italy
e-mail: cristina.cova@mail.polimi.it

L. Sala
Politecnico di Milano, Piazza Leonardo da Vinci 32, I-20133 Milano, Italy

Université de Strasbourg, Laboratoire IRMA, 7, rue René Descartes, 67084 Strasbourg Cedex, France
e-mail: lorenzo3.sala@mail.polimi.it

A. Veneziani
Department of Mathematics and Computer Science, Emory University, Atlanta, GA 30322, USA
e-mail: ale@mathcs.emory.edu

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_15

cedures (or, more in general, control problems), it is crucial to have efficient model reduction techniques to rapidly solve, for instance, a parametrized problem for several choices of the parameters of interest. In this work, we present some preliminary results merging Hi-Mod techniques with a classical Proper Orthogonal Decomposition (POD) strategy. We name this new approach as Hi-POD model reduction. We demonstrate the efficiency and the reliability of Hi-POD on multi-parameter advection-diffusion-reaction problems as well as on the incompressible Navier-Stokes equations, both in a steady and in an unsteady setting.

15.1 Introduction

The growing request of efficient and reliable numerical simulations for modeling, designing and optimizing engineering systems in a broad sense, challenges traditional methods for solving partial differential equations (PDEs). While general purpose methods like finite elements are suitable for high fidelity solutions of direct problems, practical applications often require to deal with multi-query settings, where the right balance between accuracy and efficiency becomes critical. Customization of methods to exploit all the possible features of the problem at hand may yield significant improvements in terms of efficiency, possibly with no meaningful loss in the accuracy required by engineering problems.

In this paper we focus on parametrized PDEs to model advection-diffusion-reaction phenomena as well as incompressible fluid dynamic problems in pipes or elongated domains. In particular, we propose to combine the Hierarchical Model (Hi-Mod) reduction technique, which is customized on problems featuring a leading dynamics triggered by the geometry, with a standard Proper Orthogonal Decomposition (POD) approach for a rapid solution of parametrized settings.

A Hi-Mod approximation represents a fluid in a pipe as a one-dimensional mainstream, locally enriched via transverse components. This separate description of dynamics leads to construct enhanced 1D models, where locally higher fidelity approximations are added to a backbone one-dimensional discretization [4, 16–18, 20]. The rationale behind a Hi-Mod approach is that a 1D classical model can be effectively improved by a spectral approximation of transverse components. In fact, the high accuracy of spectral methods guarantees, in general, that a low number of modes suffices to obtain a reliable approximation, yet with contained computational costs.

POD is a popular strategy in design, assimilation and optimization contexts, and relies on the so-called offline-online paradigm [6, 8, 10, 24, 28]. The offline stage computes the (high fidelity) solution to the problem at hand for a set of samples of the selected parameters. Then, an educated basis (called POD basis) is built by optimally extracting the most important components of the offline solutions (called snapshots), collected in the so-called response matrix, via a singular value decomposition. Finally, in the online phase, the POD basis is used to efficiently represent the solution associated with new values of the parameters of interest, *a priori* unknown.

In the Hi-POD procedure, the Hi-Mod reduction is used to build the response matrix during the offline stage. Then, we perform the online computation by assembling the Hi-Mod matrix associated with the new parameter and, successively, by projecting such a matrix onto the POD basis. As we show in this work, Hi-POD demonstrates to be quite competitive on a set of multiparameter problems, including linear scalar advection-diffusion-reaction problems and the incompressible Navier-Stokes equations.

The paper is organized as follows. In Sect. 15.2, we detail the Hi-POD technique and we apply it to an advection-diffusion-reaction problem featuring six parameters, pinpointing the efficiency of the procedure. Section 15.3 generalizes Hi-POD to a vector problem, by focusing on the steady incompressible Navier-Stokes equations, while the unsteady case is covered in Sect. 15.4. Some conclusions are drawn in Sect. 15.5, where some hints for a possible future investigation are also provided.

15.2 Hi-POD Reduction of Parametrized PDEs: Basics

Merging of Hi-Mod and POD procedures for parametrized PDEs has been proposed in [12, 13], in what we called *Hi-POD method*. We briefly recall the two ingredients, separately. Then, we illustrate a basic example of Hi-POD technique.

15.2.1 The Hi-Mod Setting

Let $\Omega \subset \mathbb{R}^d$ be a d-dimensional domain, with $d = 2, 3$, that makes sense to represent as $\Omega \equiv \bigcup_{x\in\Omega_{1D}}\{x\} \times \Sigma_x$, where Ω_{1D} is the $1D$ horizontal supporting fiber, while $\Sigma_x \subset \mathbb{R}^{d-1}$ represents the transverse section at $x \in \Omega_{1D}$. The reference morphology is a pipe, where the dominant dynamics occurs along Ω_{1D}. We generically consider an elliptic problem in the form

$$\text{find } u \in V : a(u, v) = F(v) \quad \forall v \in V, \tag{15.1}$$

where $V \subseteq H^1(\Omega)$ is a Hilbert space, $a(\cdot, \cdot) : V \times V \to \mathbb{R}$ is a coercive, continuous bilinear form and $F(\cdot) : V \to \mathbb{R}$ is a linear and continuous form. Standard notation for the function spaces is adopted [11]. We refer to u in (15.1) as to the *full solution*. The solution to this problem is supposed to depend on some parameters that we will highlight in our notation later on.

In the Hi-Mod reduction procedure, we introduce the space

$$V_m^h = \Big\{ v_m^h(x, \mathbf{y}) = \sum_{k=1}^{m} \tilde{v}_k^h(x)\varphi_k(\mathbf{y}), \text{ with } \tilde{v}_k^h \in V_{1D}^h,\ x \in \Omega_{1D},\ \mathbf{y} \in \Sigma_x \Big\},$$

where $V_{1D}^h \subset H^1(\Omega_{1D})$ is a discrete space of size N_h, $\{\varphi_k\}_{k\in\mathbb{N}^+}$ is a basis of L^2-orthonormal modal functions (independent of x) to describe the dynamics in Σ_x, for x varying along Ω_{1D}. For more details about the choice of the modal basis, we refer to [2, 14, 20], while V_{1D}^h may be a classical finite element space [4, 17, 18, 20] or an isogeometric function space [21].

The modal index $m \in \mathbb{N}^+$ determines the level of detail of the Hi-Mod reduced model. It may be fixed *a priori*, driven by some preliminary knowledge of the phenomenon at hand as in [4, 20], or automatically chosen via an *a posteriori* modeling error analysis as in [17, 19]. Index m can be varied along the domain to better capture local dynamics [18, 19]. For simplicity, here we consider m to be given and constant along the whole domain (*uniform* Hi-Mod reduction).

For a given modal index $m \in \mathbb{N}^+$, the Hi-Mod formulation reads as

$$\text{find } u_m^h \in V_m^h : a(u_m^h, v_m^h) = F(v_m^h) \quad \forall v_m^h \in V_m^h. \tag{15.2}$$

The well-posedness of formulation (15.2) as well as the convergence of u_m^h to u can be proved under suitable assumptions on space V_m^h [20].

In particular, after denoting by $\{\vartheta_j\}_{j=1}^{N_h}$ a basis of the space V_{1D}^h, for each element $v_m^h \in V_m^h$, the Hi-Mod expansion reads

$$v_m^h(x, \mathbf{y}) = \sum_{k=1}^{m} \Big[\sum_{j=1}^{N_h} \tilde{v}_{k,j}\vartheta_j(x) \Big] \varphi_k(\mathbf{y}).$$

The unknowns of (15.2) are the mN_h coefficients $\big\{\tilde{u}_{k,j}\big\}_{j=1,k=1}^{N_h,m}$ identifying the Hi-Mod solution u_m^h. The Hi-Mod reduction obtains a system of m coupled "psychologically" 1D problems. For m small (i.e., when the mainstream dominates the dynamics), the solution process competes with purely 1D numerical models. Accuracy of the model can be improved locally by properly setting m. From an algebraic point of view, we solve the linear system $A_m^h \mathbf{u}_m^h = \mathbf{f}_m^h$, where $A_m^h \in \mathbb{R}^{mN_h \times mN_h}$ is the Hi-Mod stiffness matrix, $\mathbf{u}_m^h \in \mathbb{R}^{mN_h}$ is the vector of the Hi-Mod coefficients and $\mathbf{f}_m^h \in \mathbb{R}^{mN_h}$ is the Hi-Mod right-hand side.

15.2.2 POD Solution of Parametrized Hi-Mod Problems

Let us denote by $\boldsymbol{\alpha}$ a vector of parameters the solution of problem (15.1) depends on. We reflect this dependence in our notation by writing the Hi-Mod solution as

$$u_m^h(\boldsymbol{\alpha}) = u_m^h(x, \mathbf{y}, \boldsymbol{\alpha}) = \sum_{k=1}^{m} \Big[\sum_{j=1}^{N_h} \tilde{u}_{k,j}^{\boldsymbol{\alpha}} \vartheta_j(x) \Big] \varphi_k(\mathbf{y}), \qquad (15.3)$$

corresponding to the algebraic Hi-Mod system

$$A_m^h(\boldsymbol{\alpha}) \mathbf{u}_m^h(\boldsymbol{\alpha}) = \mathbf{f}_m^h(\boldsymbol{\alpha}). \qquad (15.4)$$

The Hi-Mod approximation to problem (15.1) will be indifferently denoted via (15.3) or by the vector $\mathbf{u}_m^h(\boldsymbol{\alpha})$.

The goal of the Hi-POD procedure that we describe hereafter is to rapidly estimate the solution to (15.1) for a specific set $\boldsymbol{\alpha}^*$ of data, by exploiting Hi-Mod solutions previously computed for different choices of the parameter vector. The rationale is to reduce the computational cost of the solution to (15.4), yet preserving reliability.

According to the POD approach, we exploit an offline/online paradigm, i.e.,

- we compute the Hi-Mod approximation associated with different samples of the parameter $\boldsymbol{\alpha}$ to build the POD reduced basis (*offline phase*);
- we compute the solution for $\boldsymbol{\alpha}^*$ by projecting system (15.4) onto the space spanned by the POD basis (*online phase*).

15.2.2.1 The Offline Phase

We generate the reduced POD basis relying on a set of available samples of the solution computed with the Hi-Mod reduction. Even though offline costs are not usually considered in evaluating the advantage of a POD procedure, also this stage may introduce a computational burden when many samples are needed, like in multiparametric problems. The generation of snapshots with the Hi-Mod approach, already demonstrated to be significantly faster [14], mitigates the costs of this phase. The pay-off of the procedure is based on the expectation that the POD basis is considerably lower-size than the order mN_h of the Hi-Mod system. We will discuss this aspect in the numerical assessment.

Let S be the so-called *response* (or *snapshot*) *matrix*, collecting L Hi-Mod solutions to (15.1), for p different values $\boldsymbol{\alpha}_i$ of the parameter, with $i = 1, \ldots, L$. Precisely, we identify each Hi-Mod solution with the corresponding vector in (15.4),

$$\mathbf{u}_m^h(\boldsymbol{\alpha}_i) = \big[\tilde{u}_{1,1}^{\boldsymbol{\alpha}_i}, \ldots, \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_i}, \tilde{u}_{2,1}^{\boldsymbol{\alpha}_i}, \ldots, \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_i}, \ldots, \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_i}\big]^T \in \mathbb{R}^{mN_h}, \qquad (15.5)$$

the unknown coefficients being ordered mode-wise. Thus, the response matrix $S \in \mathbb{R}^{(mN_h)\times L}$ reads

$$S = \left[\mathbf{u}_m^h(\boldsymbol{\alpha}_1), \mathbf{u}_m^h(\boldsymbol{\alpha}_2), \ldots, \mathbf{u}_m^h(\boldsymbol{\alpha}_L)\right] = \begin{bmatrix} \tilde{u}_{1,1}^{\boldsymbol{\alpha}_1} & \tilde{u}_{1,1}^{\boldsymbol{\alpha}_2} & \ldots & \tilde{u}_{1,1}^{\boldsymbol{\alpha}_L} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_1} & \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_2} & \ldots & \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_L} \\ \tilde{u}_{2,1}^{\boldsymbol{\alpha}_1} & \tilde{u}_{2,1}^{\boldsymbol{\alpha}_2} & \ldots & \tilde{u}_{2,1}^{\boldsymbol{\alpha}_L} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_1} & \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_2} & \ldots & \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_L} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_1} & \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_2} & \ldots & \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_L} \end{bmatrix}. \qquad (15.6)$$

The selection of representative values of the parameter is clearly critical in the effectiveness of the POD procedure. More the snapshots cover the entire parameter space and more evident the model reduction will be. This is a nontrivial issue, generally problem dependent. For instance, in [9] the concept of *domain of effectiveness* is introduced to formalize the region of the parameter space accurately covered by a snapshot in a problem of cardiac conductivity. In this preliminary work, we do not dwell with this aspect since we work on more general problems. A significant number of snapshots is anyhow needed to construct an efficient POD basis, the Hi-Mod procedure providing an effective tool for this purpose (with respect to a full finite element generation of the snapshots).

To establish a correlation between the POD procedure and statistical moments, we enforce the snapshot matrix to have null average by setting

$$R = S - \frac{1}{L}\sum_{i=1}^{L} \begin{bmatrix} \tilde{u}_{1,1}^{\boldsymbol{\alpha}_i} & \tilde{u}_{1,1}^{\boldsymbol{\alpha}_i} & \ldots & \tilde{u}_{1,1}^{\boldsymbol{\alpha}_i} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_i} & \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_i} & \ldots & \tilde{u}_{1,N_h}^{\boldsymbol{\alpha}_i} \\ \tilde{u}_{2,1}^{\boldsymbol{\alpha}_i} & \tilde{u}_{2,1}^{\boldsymbol{\alpha}_i} & \ldots & \tilde{u}_{2,1}^{\boldsymbol{\alpha}_i} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_i} & \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_i} & \ldots & \tilde{u}_{2,N_h}^{\boldsymbol{\alpha}_i} \\ \vdots & \vdots & \vdots & \vdots \\ \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_i} & \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_i} & \ldots & \tilde{u}_{m,N_h}^{\boldsymbol{\alpha}_i} \end{bmatrix} \in \mathbb{R}^{(mN_h)\times L}. \qquad (15.7)$$

By Singular Value Decomposition (SVD), we write

$$R = \Psi \Sigma \Phi^T,$$

with $\Psi \in \mathbb{R}^{(mN_h)\times(mN_h)}$, $\Sigma \in \mathbb{R}^{(mN_h)\times L}$, $\Phi \in \mathbb{R}^{L\times L}$. Matrices Ψ and Φ are unitary and collect the left and the right singular vectors of R, respectively. Matrix $\Sigma =$ diag $(\sigma_1, \ldots, \sigma_q)$ is pseudo-diagonal, $\sigma_1, \sigma_2, \ldots, \sigma_q$ being the singular values of R, with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_q$ and $q = \min\{mN_h, L\}$ [5]. In the numerical assessment below, we take $q = L$.

The POD (orthogonal) basis is given by the l left singular vectors $\{\boldsymbol{\psi}_i\}$ associated with the most significant l singular values, with $l \ll mN_h$. Different criteria can be pursued to select those singular values. A possible approach is to select the first l ordered singular values, such that $\sum_{i=1}^{l} \sigma_i^2 / \sum_{i=1}^{q} \sigma_i^2 \geq \varepsilon$ for a positive user-defined tolerance ε [28]. The reduced POD space then reads $V_{\text{POD}}^l = \text{span}\{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_l\}$, with $\dim(V_{\text{POD}}^l) = l$.

Equivalently, we can identify the POD basis by applying the spectral decomposition to the covariance matrix $C \equiv R^T R$ (being $mN_h \geq L$). As well known, the right singular vectors of R coincide with the eigenvectors $\mathbf{c}_i$ of C, with eigenvalues $\lambda_i = \sigma_i^2$, for $i = 1, \ldots, L$. Thus, the POD basis functions reads $\boldsymbol{\psi}_i = \lambda_i^{-1} S \mathbf{c}_i$ [28].

15.2.2.2 The Online Phase

We aim at rapidly computing the Hi-Mod approximation to problem (15.1) for the parameter value $\boldsymbol{\alpha}^*$ not included in the sampling set $\{\boldsymbol{\alpha}_i\}_{i=1}^{L}$. For this purpose, we assume an affine parameter dependence. Then, we project the Hi-Mod system (15.4), with $\boldsymbol{\alpha} = \boldsymbol{\alpha}^*$, onto the POD space V_{POD}^l, by solving the linear system

$$A_{\text{POD}}^{\boldsymbol{\alpha}^*} \mathbf{u}_{\text{POD}}^{\boldsymbol{\alpha}^*} = \mathbf{f}_{\text{POD}}^{\boldsymbol{\alpha}^*},$$

with $A_{\text{POD}}^{\boldsymbol{\alpha}^*} = (\Psi_{\text{POD}}^l)^T A_m^h(\boldsymbol{\alpha}^*)\, \Psi_{\text{POD}}^l \in \mathbb{R}^{l \times l}$, $\mathbf{f}_{\text{POD}}^{\boldsymbol{\alpha}^*} = (\Psi_{\text{POD}}^l)^T \mathbf{f}_m^h(\boldsymbol{\alpha}^*) \in \mathbb{R}^l$ and $\mathbf{u}_{\text{POD}}^{\boldsymbol{\alpha}^*} = [u_{\text{POD},1}^{\boldsymbol{\alpha}^*}, \ldots, u_{\text{POD},l}^{\boldsymbol{\alpha}^*}]^T \in \mathbb{R}^l$, where $A_m^h(\boldsymbol{\alpha}^*)$ and $\mathbf{f}_m^h(\boldsymbol{\alpha}^*)$ are defined as in (15.4), and $\Psi_{\text{POD}}^l = [\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_l] \in \mathbb{R}^{(mN_h) \times l}$ is the matrix collecting, by column, the POD basis functions.

By exploiting the POD basis, we write

$$\mathbf{u}_m^h(\boldsymbol{\alpha}^*) \approx \sum_{s=1}^{l} u_{\text{POD},s}^{\boldsymbol{\alpha}^*} \boldsymbol{\psi}_s.$$

The construction of $A_{\text{POD}}^{\boldsymbol{\alpha}^*}$ and $\mathbf{f}_{\text{POD}}^{\boldsymbol{\alpha}^*}$ requires the assembly of the Hi-Mod matrix/right-hand side for the value $\boldsymbol{\alpha}^*$, successively projected onto the POD space. Also in the basic POD online phase, we need to assembly, in general, the full problem, and the Hi-Mod model, featuring lower size than a full finite element problem, gives a computational advantage. In addition, the final solution is computed by solving an $l \times l$ system as opposed to the $mN_h \times mN_h$ Hi-Mod system, with a clear overall computational advantage, as we verify hereafter. In absence of an affine parameter dependence, we can resort to an empirical interpolation method as explained, e.g., in [24].

15.2.3 Numerical Assessment

In this preliminary paper, we consider only 2D problems, the 3D case being a development of the present work. We consider the linear advection-diffusion-reaction (ADR) problem

$$\begin{cases} -\nabla \cdot \big(\mu(\mathbf{x})\nabla u(\mathbf{x})\big) + \mathbf{b}(\mathbf{x}) \cdot \nabla u(\mathbf{x}) + \sigma(\mathbf{x})u(\mathbf{x}) = f(\mathbf{x}) & \text{in } \Omega \\ u(\mathbf{x}) = 0 & \text{on } \Gamma_D \\ \mu(\mathbf{x})\dfrac{\partial u}{\partial n}(\mathbf{x}) = 0 & \text{on } \Gamma_N, \end{cases} \tag{15.8}$$

with $\Gamma_D, \Gamma_N \subset \partial\Omega$, such that $\Gamma_D \cup \Gamma_N = \partial\Omega$ and $\overset{\circ}{\Gamma}_D \cap \overset{\circ}{\Gamma}_N = \emptyset$, where μ, $\mathbf{b}$, σ and f denote the viscosity, the advective field, the reactive coefficient and the source term, respectively. In particular, we set $\Omega = (0,6) \times (0,1)$, with $\Gamma_N = \{(x,y) : x = 6, 0 \leq y \leq 1,\}$ and $\Gamma_D = \partial\Omega \setminus \Gamma_N$. We also assume constant viscosity and reaction, i.e., we pick $\mu = 0.1\mu_0$ for $\mu_0 \in [1,10]$ and $\sigma \in [0,3]$; then, we assign a sinusoidal advective field, $\mathbf{b}(\mathbf{x}) = [b_1, b_2\sin(6x)]^T$ with $b_1 \in [2,20]$ and $b_2 \in [1,3]$, and the source term $f(\mathbf{x}) = f_1\chi_{C_1}(\mathbf{x}) + f_2\chi_{C_2}(\mathbf{x})$ for f_1, $f_2 \in [5,25]$ and where function χ_ω denotes the characteristic function associated with the generic domain ω, $C_1 = \{(x,y) : (x-1.5)^2 + 0.4\,(y-0.25)^2 < 0.01\}$ and $C_2 = \{(x,y) : (x-0.75)^2 + 0.4\,(y-0.75)^2 < 0.01\}$ identifying two ellipsoidal areas in Ω. According to the notation in (15.1), we set therefore $V \equiv H^1_{\Gamma_D}(\Omega)$, $a(u,v) \equiv \big(\mu\nabla u, \nabla v\big) + \big(\mathbf{b}\cdot\nabla u + \sigma u, v\big)$, for any $u, v \in V$, and $F(v) = \big(f, v\big)$, for any $v \in V$, $\big(\cdot,\cdot\big)$ denoting the $L^2(\Omega)$-scalar product.

In the offline phase, we select $L = 30$ problems, by *randomly* varying coefficients $\mu_0, \sigma, b_1, b_2, f_1$ and f_2 in the corresponding ranges, so that $\boldsymbol{\alpha} \equiv [\mu_0, \sigma, b_1, b_2, f_1, f_2]^T$. We introduce a uniform partition of Ω_{1D} into 121 sub-intervals, and we Hi-Mod approximate the selected L problems, combining piecewise linear finite elements along the 1D fiber with a modal expansion based on 20 sinusoidal functions along the transverse direction.

In the online phase, we aim at computing the Hi-Mod approximation to problem (15.8) for $\boldsymbol{\alpha} = \boldsymbol{\alpha}^* = [\mu_0^*, \sigma^*, b_1^*, b_2^*, f_1^*, f_2^*]^T$, with

$$\mu_0^* = 2.4, \quad \sigma^* = 0, \quad b_1^* = 5, \quad b_2^* = 1, \quad f_1^* = f_2^* = 10.$$

Figure 15.1 shows a Hi-Mod reference solution, $u_m^{R,h}$, computed by directly applying Hi-Mod reduction to (15.8) for $\boldsymbol{\alpha} = \boldsymbol{\alpha}^*$, with the same Hi-Mod discretization setting used for the offline phase.

This test is intended to demonstrate the reliability of Hi-POD to construct an approximation of the Hi-Mod solution (that, in turn, approximates the full solution u), with a contained computational cost.

Figure 15.2 shows the spectrum of the response matrix R in (15.7). As highlighted by the vertical lines, we select four different values for the number l of POD modes, i.e., $l = 2, 6, 19, 29$. For these choices, the ratio $\sum_{i=1}^{l}\sigma_i^2 / \sum_{i=1}^{q}\sigma_i^2$ assumes the value 0.780 for $l = 2$, 0.971 for $l = 6$, 0.999 for $l = 19$ (and, clearly, 1 for $l = 29$).

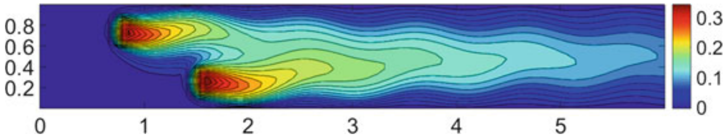


Fig. 15.1 ADR problem. Hi-Mod reference solution

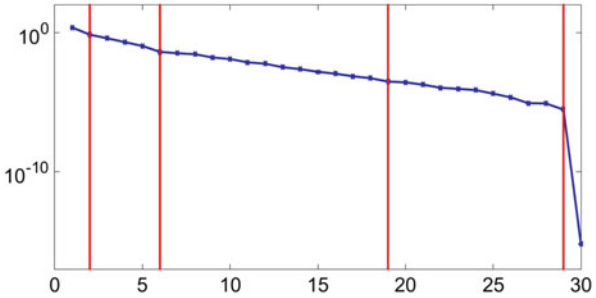


Fig. 15.2 ADR problem. Singular values of the response matrix *R*

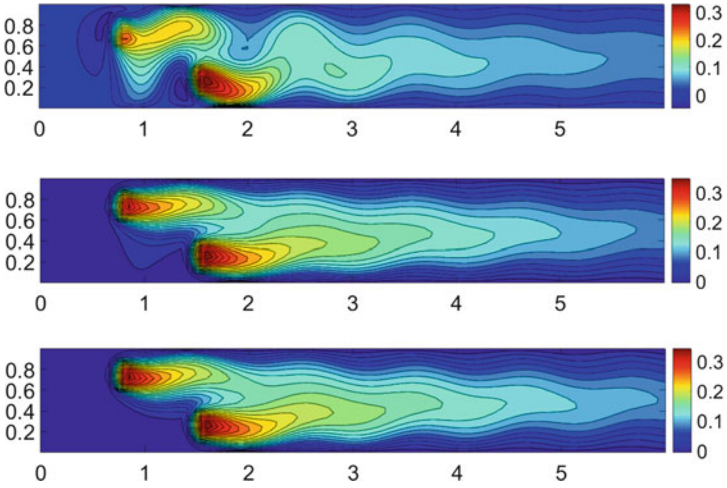


Fig. 15.3 ADR problem. Hi-Mod approximation provided by the Hi-POD approach for $l = 2$ (*top*), $l = 6$ (*center*), $l = 19$ (*bottom*)

The singular values for the specific problem decay quite slowly. This is due to the presence of many (six) parameters, so that the redundancy of the snapshots (that triggers the decay) is quite limited.

Nevertheless, we observe that the Hi-POD solution still furnishes a reliable and rapid approximation of the solution in correspondence of the value $\boldsymbol{\alpha}^*$. Precisely, Fig. 15.3 shows the Hi-Mod approximation provided by Hi-POD, for $l = 2, 6, 19$ (top-bottom). We stress that six POD modes are enough to obtain a Hi-Mod reduced

Table 15.1 ADR problem

	$l = 2$	$l = 6$	$l = 19$	$l = 29$
$\frac{\|\|u_m^{R,h} - u_m^h(\boldsymbol{\alpha}^*)\|\|_{L^2(\Omega)}}{\|\|u_m^{R,h}\|\|_{L^2(\Omega)}}$	3.52e-01	3.44e-02	9.71e-04	4.38e-04
$\frac{\|\|u_m^{R,h} - u_m^h(\boldsymbol{\alpha}^*)\|\|_{H^1(\Omega)}}{\|\|u_m^{R,h}\|\|_{H^1(\Omega)}}$	4.54e-01	6.88e-02	2.21e-03	8.24e-04

Relative errors for different Hi-POD reconstructions of the Hi-Mod solution

solution which, qualitatively, exhibits the same features as $u_m^{R,h}$. Moreover, the contribution of singular vectors for $l > 19$ is of no improvement. We also notice that the results for $l = 6$ are excellent, in spite of the large number of parameters.

Table 15.1 provides more quantitative information. We collect the $L^2(\Omega)$- and the $H^1(\Omega)$-norm of the relative error obtained by replacing the Hi-Mod reference solution with the one provided by the Hi-POD approach. As expected, the error diminishes as the number of POD modes increases.

15.3 Hi-POD Reduction of the Navier-Stokes Equations

We generalize the Hi-POD procedure in Sect. 15.2.2 to the incompressible Navier-Stokes equations [25]. We first consider the stationary problem

$$\begin{cases} -\nabla \cdot (2\nu\, \mathbb{D}(\mathbf{u}))\,(\mathbf{x}) + (\mathbf{u} \cdot \nabla)\,\mathbf{u}(\mathbf{x}) + \nabla p(\mathbf{x}) = \mathbf{f}(\mathbf{x}) & \text{in } \Omega \\ \nabla \cdot \mathbf{u}(\mathbf{x}) = 0 & \text{in } \Omega \\ \mathbf{u}(\mathbf{x}) = \mathbf{0} & \text{on } \Gamma_D \\ (\mathbb{D}(\mathbf{u}) - p\mathbb{I})\,(\mathbf{x})\,\mathbf{n} = g\mathbf{n} & \text{on } \Gamma_N, \end{cases} \tag{15.9}$$

with $\mathbf{u} = [u_1, u_2]^T$ and p the velocity and the pressure of the flow, respectively $\nu > 0$ the kinematic viscosity, $\mathbb{D}(\mathbf{u}) = \frac{1}{2}\left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)$ the strain rate, $\mathbf{f}$ the force per unit mass, $\mathbf{n}$ the unit outward normal vector to the domain boundary $\partial\Omega$, $\mathbb{I}$ the identity tensor, g a sufficiently regular function, and where Γ_D and Γ_N are defined as in (15.8). We apply a standard Picard linearization of the nonlinear term

$$\begin{cases} -\nabla \cdot \left(2\nu\, \mathbb{D}(\mathbf{u}^{k+1})\right) + \left(\mathbf{u}^k \cdot \nabla\right) \mathbf{u}^{k+1} + \nabla p^{k+1} = \mathbf{f} & \text{in } \Omega \\ \nabla \cdot (\mathbf{u}^{k+1}) = 0 & \text{in } \Omega \\ \mathbf{u}^{k+1} = \mathbf{0} & \text{on } \Gamma_D \\ \left(\mathbb{D}(\mathbf{u}^{k+1}) - p^{k+1}\mathbb{I}\right)\mathbf{n} = g\mathbf{n} & \text{on } \Gamma_N, \end{cases}$$

where $\{\mathbf{u}^j, p^j\}$ denotes the unknown pair at the iteration j. Stopping criterion of the Picard iteration is designed on the increment between two consecutive iterations.

Problem (15.9) is approximated via a standard Hi-Mod technique, for both the velocity and the pressure, where a modal basis constituted by orthogonal Legendre polynomials, adjusted to include the boundary conditions, is used. Finite elements are used along the centerline. The finite dimension Hi-Mod spaces for velocity and pressure obtained by the combination of different discretization methods need to be inf-sup compatible. Unfortunately, no proof of compatibility is currently available, even though some empirical strategies based on the Bathe-Chapelle test are available [7, 14]. In particular, here we take piecewise quadratic velocity/linear pressure along the mainstream and the numbers $m_p, m_{\mathbf{u}}$ of pressure and velocity modes is set such that $m_{\mathbf{u}} = m_p + 2$. Numerical evidence suggests this to be an inf-sup compatible choice [1, 7]. Finally, the same number of modes is used for the two velocity components, for the sake of simplicity.

We denote by $V_{1D}^{h,u} \subset H^1(\Omega_{1D})$ and by $V_{1D}^{h,p} \subset L^2(\Omega_{1D})$ the finite element space adopted to discretize u_1, u_2 and p, respectively along Ω_{1D}, with $\dim(V_{1D}^{h,u}) = N_{h,u}$ and $\dim(V_{1D}^{h,p}) = N_{h,p}$. Thus, the total number of degrees of freedom involved by a Hi-Mod approximation of $\mathbf{u}$ and p is $N_{\mathbf{u}} = 2m_{\mathbf{u}}N_{h,u}$ and $N_p = m_pN_{h,p}$, respectively.

From an algebraic viewpoint, at each Picard iteration, we solve the linear system (we omit index k for easiness of notation)

$$S^h_{\{m_{\mathbf{u}},m_p\}}\,\mathbf{z}^h_{m_{\mathbf{u}},m_p} = \mathbf{F}^h_{\{m_{\mathbf{u}},m_p\}}, \tag{15.10}$$

where

$$S^h_{\{m_{\mathbf{u}},m_p\}} = \begin{bmatrix} C^h_{\{m_{\mathbf{u}},m_{\mathbf{u}}\}} & [B^h_{\{m_{\mathbf{u}},m_p\}}]^T \\ B^h_{\{m_{\mathbf{u}},m_p\}} & 0 \end{bmatrix} \in \mathbb{R}^{(N_{\mathbf{u}}+N_p)\times(N_{\mathbf{u}}+N_p)},$$

with $C^h_{\{m_{\mathbf{u}},m_{\mathbf{u}}\}} \in \mathbb{R}^{N_{\mathbf{u}}\times N_{\mathbf{u}}}$, $B^h_{\{m_{\mathbf{u}},m_p\}} \in \mathbb{R}^{N_p\times N_{\mathbf{u}}}$ the Hi-Mod momentum and divergence matrix, respectively, $\mathbf{z}^h_{m_{\mathbf{u}},m_p} = [\mathbf{u}^h_{m_{\mathbf{u}}}, p^h_{m_p}]^T \in \mathbb{R}^{N_{\mathbf{u}}+N_p}$ the vector of the Hi-Mod solutions, and where $\mathbf{F}^h_{\{m_{\mathbf{u}},m_p\}} = [\mathbf{f}^h_{m_{\mathbf{u}}}, \mathbf{0}]^T \in \mathbb{R}^{N_{\mathbf{u}}+N_p}$, with $\mathbf{f}^h_{m_{\mathbf{u}}}$ the Hi-Mod right-hand side of the momentum equation.

When coming to the Hi-POD procedure for problem (15.9), we follow a segregated procedure, where a basis function set is constructed for the velocity and another one for the pressure. The effectiveness of this reduced basis in representing the solution for a different value of the parameter is higher with respect to a monolithic approach, where a unique POD basis is built. We will support this statement with numerical evidence. Still referring to (15.6) and (15.7), we build two separate response matrices, $R_{\mathbf{u}} \in \mathbb{R}^{N_{\mathbf{u}}\times L}$ and $R_p \in \mathbb{R}^{N_p\times L}$, which gather, by column, the Hi-Mod approximation for the velocity, $\mathbf{u}^h_{m_{\mathbf{u}}}(\boldsymbol{\alpha}) \in \mathbb{R}^{N_{\mathbf{u}}}$, and for the pressure, $p^h_{m_p}(\boldsymbol{\alpha}) \in \mathbb{R}^{N_p}$, solutions to the Navier-Stokes problem (15.9) for L different choices $\boldsymbol{\alpha}_i$, with $i = 1, \ldots, L$, of the parameter that, in this case, is $\boldsymbol{\alpha} = [\nu, \mathbf{f}, g]^T$. A standard block-Gaussian procedure resorting to the pressure Schur-complement is used to compute velocity and pressure, separately [3].

Following a segregated SVD analysis of the two unknowns, after identifying the two indices l_u and l_p, separately, we construct a unique reduced POD space V^l_{POD}, with $l = \max(l_u, l_p)$, by collecting the first l singular vectors of $R_{\mathbf{u}}$ and of R_p. More precisely, for a new value $\boldsymbol{\alpha}^*$ of the parameters, with $\boldsymbol{\alpha}^* \neq \boldsymbol{\alpha}_i$ for $i = 1, \ldots, L$, at each Picard iteration, we project the linearized Navier-Stokes problem onto the space V^l_{POD}.

Another possible approach is to keep the computation of the velocity and pressure separate on the two basis function sets with size l_u and l_p, by resorting to an approximation of the pressure Schur complement, followed by the computation of the velocity, similar to what is done in algebraic splittings [3, 22, 26, 27]. More in general, the treatment of the nonlinear term in the Navier-Stokes problem can follow approximation strategies with a specific basis function set and *empirical interpolation strategies* [24]. At this preliminary stage, we do not follow this approach and we just assess the performances of the basic procedure. However, this topic will be considered in the follow-up of the present work in view of real applications.

It is also worth noting that no inf-sup compatibility is guaranteed for the POD basis functions. Numerical evidence suggests that we do have inf-sup compatible basis functions, however a theoretical analysis is still missing.

15.3.1 A Benchmark Test Case

We solve problem (15.9) on the rectangular domain $\Omega = (0, 8) \times (-2, 2)$, where $\Gamma_D = \{(x, y) \,:\, 0 \leq x \leq 8, y = \pm 2\}$ and $\Gamma_N = \partial\Omega \setminus \Gamma_D$.

Moreover, we assume the analytical representation

$$\mathbf{f} = \begin{bmatrix} f_{0,x} + f_{xx}\, x + f_{xy}\, y \\ f_{0,y} + f_{yx}\, x + f_{yy}\, y \end{bmatrix} \tag{15.11}$$

for the forcing term $\mathbf{f}$ involved in the parameter $\boldsymbol{\alpha}$.

In the offline stage, we Hi-Mod approximate $L = 30$ problems, by varying the coefficients f_{st}, for $s = 0, x, y$ and $t = x, y$, in (15.11), the kinematic viscosity ν and the boundary value g in (15.9). In particular, we randomly sample the coefficients f_{st} on the interval $[0, 100]$, whereas we adopt a uniform sampling for ν on $[30, 70]$ and for g on $[1, 80]$.

Concerning the adopted Hi-Mod discretization, we partition the fiber Ω_{1D} into 80 uniform sub-intervals to employ quadratic and linear finite elements for the velocity and the pressure, respectively. Five Legendre polynomials are used to describe the transverse trend of $\mathbf{u}$, while three modal functions are adopted for p.

In the online phase, we compute the Hi-POD approximation to problem (15.9) with parameters $\boldsymbol{\alpha}^* = [\mathbf{f}^*, \nu^*, g^*]^T$, with $\mathbf{f}^* = [82.6, 12.1]^T$, $\nu^* = 51.4$ and $g^* = 24.2, f_{xx} = f_{yy} = f_{xy} = f_{yx} = 0$. Figure 15.4 (left) shows the contour plots of the two components of the velocity and of the pressure for the reference Hi-Mod solution $\{\mathbf{u}_{m_\mathbf{u}}^{R,h}, p_{m_p}^{R,h}\}$ (from top to bottom: horizontal velocity, vertical velocity, pressure), with $\mathbf{u}_{m_\mathbf{u}}^{R,h} = [u_{m_\mathbf{u},1}^{R,h}, u_{m_\mathbf{u},2}^{R,h}]^T$.

For the sake of completeness, we display the results of a monolithic approach in Fig. 15.4 (center and right), where the POD basis is computed on a unique response matrix for the velocity and pressure. While velocity results are quite accurate, pressure approximation is bad, suggesting that, probably, a lack of inf-sup compatibility of the reduced basis leads to unreliable pressure approximations, independently of the dimension of the POD space.

When we turn to the segregated approach, Fig. 15.5 shows the distribution of the singular values of the response matrices $R_\mathbf{u}$ and R_p, respectively. Again the values decay is not so rapid to pinpoint a clear cut-off value (at least for significantly small dimensions of the reduced basis), as a consequence of the multiple parametrization that inhibits the redundancy of the snapshots. However, when we compare the Hi-Mod solution identified by three different choices of the POD spaces, $V_{\text{POD}}^{l,\mathbf{u}}$

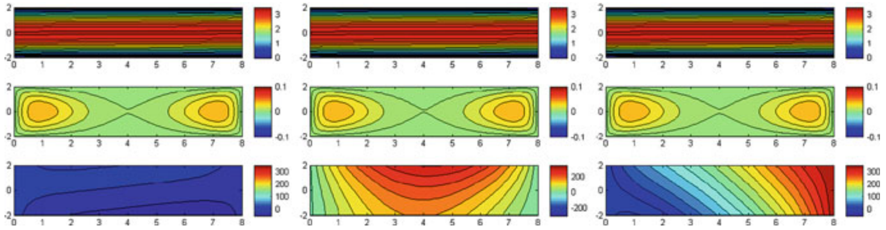


Fig. 15.4 Steady Navier-Stokes equations. Hi-Mod reference solution (*left*), Hi-Mod approximation yielded by the monolithic Hi-POD approach for $l = 11$ (*center*) and $l = 28$ (*right*): horizontal (*top*) and vertical (*middle*) velocity components; pressure (*bottom*)

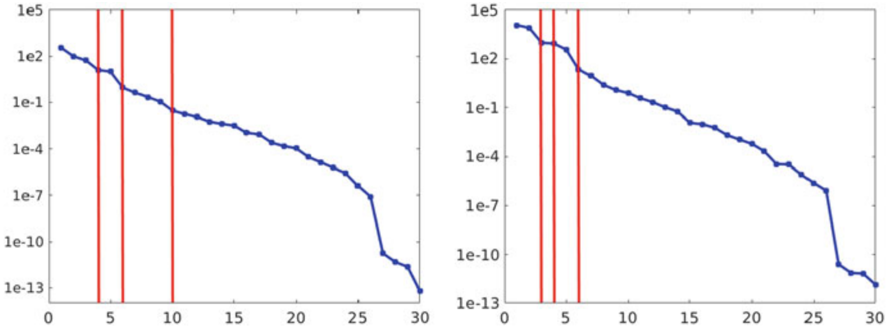


Fig. 15.5 Steady Navier-Stokes equations. Singular values of the response matrix $R_\mathbf{u}$ (*left*) and R_p (*right*)

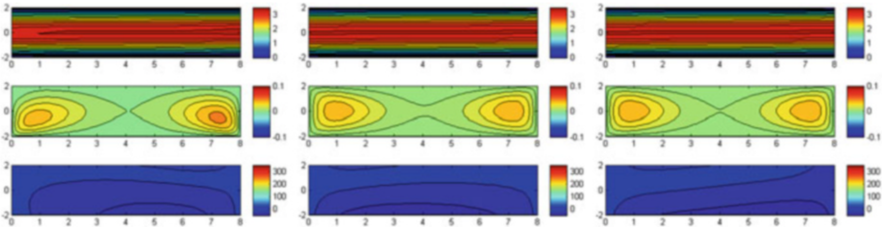


Fig. 15.6 Steady Navier-Stokes equations. Hi-POD approximation yielded by the segregated Hi-POD approach for $l = 4$ (*left*), $l = 6$ (*center*), $l = 10$ (*right*): horizontal (*top*) and vertical (*middle*) velocity components; pressure (*bottom*)

Table 15.2 Steady Navier-Stokes equations

	$\dfrac{\|\|u^{R,h}_{m_{\mathbf{u}},1} - u^h_{m_{\mathbf{u}},1}(\boldsymbol{\alpha}^*)\|\|_{H^1(\Omega)}}{\|\|u^{R,h}_{m_{\mathbf{u}},1}\|\|_{H^1(\Omega)}}$	$\dfrac{\|\|u^{R,h}_{m_{\mathbf{u}},2} - u^h_{m_{\mathbf{u}},2}(\boldsymbol{\alpha}^*)\|\|_{H^1(\Omega)}}{\|\|u^{R,h}_{m_{\mathbf{u}},2}\|\|_{H^1(\Omega)}}$	$\dfrac{\|\|p^{R,h}_{m_p} - p^h_{m_p}(\boldsymbol{\alpha}^*)\|\|_{L^2(\Omega)}}{\|\|p^{R,h}_{m}\|\|_{L^2(\Omega)}}$
$l = 4$	$7.1 \cdot 10^{-3}$	$3.9 \cdot 10^{-1}$	$4.8 \cdot 10^{-1}$
$l = 6$	$3.8 \cdot 10^{-4}$	$4.3 \cdot 10^{-2}$	$3.9 \cdot 10^{-1}$
$l = 10$	$1.1 \cdot 10^{-4}$	$8.6 \cdot 10^{-3}$	$1.3 \cdot 10^{-3}$

Relative errors for different Hi-POD reconstructions of the Hi-Mod solution

and $V^{l,p}_{\text{POD}}$, with the reference approximation in Fig. 15.4 (left), we notice that the choice $l = 6$ is enough for a reliable reconstruction of the approximate solution (see Fig. 15.6 (center)). The horizontal velocity component—being the most predominant dynamics—is captured even with a lower size of the reduced spaces $V^{l,\mathbf{u}}_{\text{POD}}$, while the pressure still represents the most challenging quantity to be correctly described.

In Table 15.2, we quantify the accuracy of the Hi-POD procedure. We compare the relative errors between the Hi-Mod reference solution $\{\mathbf{u}^{R,h}_{m_{\mathbf{u}}}, p^{R,h}_{m_p}\}$ and the Hi-POD approximation $\{\mathbf{u}^h_{m_{\mathbf{u}}}(\boldsymbol{\alpha}^*), p^h_{m_p}(\boldsymbol{\alpha}^*)\}$ generated by different Hi-POD schemes, with $\mathbf{u}^h_{m_{\mathbf{u}}}(\boldsymbol{\alpha}^*) = [u^h_{m_{\mathbf{u}},1}(\boldsymbol{\alpha}^*), u^h_{m_{\mathbf{u}},2}(\boldsymbol{\alpha}^*)]^T$.

As for the computational time (in seconds),[1] we found that the segregated Hi-POD requires $0.13s$ to be compared with $0.9s$ demanded by the standard Hi-Mod approximation. This highlights the significant computational advantage attainable by Hi-POD, in particular for a rapid approximation of the incompressible Navier-Stokes equations when estimating one or more parameters of interest.

[1] All the experiments have been performed using MATLAB® R2010a 64-bit on a Fujitsu Lifebook T902 equipped with a 2.70 GHz i5 (3rd generation) vPro processor and 8 GB of RAM.

15.4 Towards More Realistic Applications

We extend the Hi-POD segregated approach to the unsteady Navier-Stokes equations

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t}(\mathbf{x},t) - \nabla \cdot (2\nu\,\mathbb{D}(\mathbf{u}))\,(\mathbf{x},t) + (\mathbf{u}\cdot\nabla)\,\mathbf{u}(\mathbf{x},t) + \nabla p(\mathbf{x},t) = \mathbf{f}(\mathbf{x},t) & \text{in } Q \\ \nabla\cdot\mathbf{u}(\mathbf{x},t) = 0 & \text{in } Q \\ \mathbf{u}(\mathbf{x},t) = \mathbf{0} & \text{on } G_D \\ (\mathbb{D}(\mathbf{u}) - p\mathbb{I})\,(\mathbf{x},t)\,\mathbf{n} = g(\mathbf{x},t)\mathbf{n} & \text{on } G_N \\ \mathbf{u}(\mathbf{x},0) = \mathbf{u}_0(\mathbf{x}) & \text{in } \Omega, \end{cases} \tag{15.12}$$

with $Q = \Omega \times I$ for $I = (0,T)$ the time window of interest, $G_D = \Gamma_D \times I$, $G_N = \Gamma_N \times I$, $\mathbf{u}_0$ the initial value, and where all the other quantities are defined as in (15.9). After introducing a uniform partition of the interval I into M sub-intervals of length Δt, we resort to the backward Euler scheme and approximate the nonlinear term via a classical first order semi-implicit scheme. The semi-discrete problem reads: for each $0 \le n \le M-1$, find $\{\mathbf{u}^{n+1}, p^{n+1}\} \in V \equiv [H^1_{\Gamma_D}(\Omega)]^2 \times L^2(\Omega)$ such that

$$\begin{cases} \dfrac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\Delta t} - \nabla\cdot\left(2\nu\,\mathbb{D}(\mathbf{u}^{n+1})\right) + (\mathbf{u}^n\cdot\nabla)\,\mathbf{u}^{n+1} + \nabla p^{n+1} = \mathbf{f}^{n+1} & \text{in } \Omega \\ \nabla\cdot\mathbf{u}^{n+1} = 0 & \text{in } \Omega \\ \mathbf{u}^{n+1} = \mathbf{0} & \text{on } \Gamma_D \\ \left(\mathbb{D}(\mathbf{u}^{n+1}) - p^{n+1}\mathbb{I}\right)\,\mathbf{n} = g^{n+1}\mathbf{n} & \text{on } \Gamma_N, \end{cases} \tag{15.13}$$

with $\mathbf{u}^0 = \mathbf{u}_0(\mathbf{x})$, $\mathbf{u}^{n+1} \simeq \mathbf{u}(\mathbf{x}, t^{n+1})$, $p^{n+1} \simeq p(\mathbf{x}, t^{n+1})$ and $t^i = i\Delta t$, for $i = 0, \dots, M$.

For the Hi-Mod approximation, we replace space V in (15.13) with the same Hi-Mod space as in the steady case.

When applied to unsteady problems, POD procedures are generally used for estimating the solution at a generic time by taking advantage of precomputed snapshots [28]. In our specific case, we know the Hi-Mod solution for a certain number of parameters $\boldsymbol{\alpha}_i$, and we aim at rapidly estimating the solution over a time interval of interest for a specific value $\boldsymbol{\alpha}^*$ of the parameter, with $\boldsymbol{\alpha}^* \neq \boldsymbol{\alpha}_i$. The procedure we propose here is the following one:

1. we precompute offline the steady Hi-Mod solution for L samples $\boldsymbol{\alpha}_i$ of the parameter, $i = 1, \dots, L$;
2. for a specific value $\boldsymbol{\alpha}^*$ of the parameter, we compute online the Hi-Mod solution to (15.12) at the first times t^j, for $j = 1, \dots, P$;
3. we juxtapose the Hi-Mod snapshots to the steady response matrix obtained offline;

4. we perform the Hi-POD procedure to estimate the solution to (15.12) at times t^j, with $j > P$.

In absence of a complete analysis of this approach, we present here some preliminary numerical results in a non-rectilinear domain. Hi-Mod reduction has been already applied to curvilinear domains [15, 21]. In particular, in [21] we exploit the isogeometric analysis to describe a curvilinear centerline Ω_{1D}, by replacing the 1D finite element discretization with an isogeometric approximation.

Here, we consider a quadrilateral domain with a sinusoidal-shaped centerline (see Fig. 15.7). We adopt the same approach as in [15] based on an affine mapping of the bent domain into a rectilinear reference one. During the offline phase, we Hi-Mod solve problem (15.9) for $L = 5$ different choices of the parameter $\boldsymbol{\alpha} = [\nu, \mathbf{f}, g]^T$, by uniformly sampling the viscosity ν in $[1.5, 7]$, g in $[1, 80]$, and $\mathbf{f}(\mathbf{x}) = [f_1, f_2]^T$, with $f_1, f_2 \in \mathbb{R}$ in $[0, 10]$. Domain Ω_{1D} is divided in 80 uniform sub-intervals. We approximate $\mathbf{u}$ and p with five and three Legendre polynomials along the transverse direction combined with piecewise quadratic and linear functions along Ω_{1D}, respectively. The corresponding Hi-Mod approximations constitute the first L columns of the response matrices $R_{\mathbf{u}}$ and R_p.

Then, we solve the unsteady problem (15.12). We pick $\mathbf{u}_0 = \mathbf{0}$, $T = 10$, and we introduce a uniform partition of the time interval I, with $\Delta t = 0.1$.

The data $\boldsymbol{\alpha}^*$ for the online phase are $\nu^* = 2.8$, $g^* = 30 + 20\sin(t)$ and $\mathbf{f}^* = [5.8, 1.1]^T$. Matrices $R_{\mathbf{u}}$ and R_p are added by the first $P = 5$ Hi-Mod approximations $\{\mathbf{u}^{h,j}_{m_{\mathbf{u}}}(\boldsymbol{\alpha}^*), p^{h,j}_{m_p}(\boldsymbol{\alpha}^*)\}$, for $j = 1, \ldots, 5$, so that $R_{\mathbf{u}} \in \mathbb{R}^{N_{\mathbf{u}} \times 10}$ and $R_p \in \mathbb{R}^{N_p \times 10}$, where $N_{\mathbf{u}} = 2 \times 5 \times N_{h,\mathbf{u}}$, $N_p = 3 \times N_{h,p}$ with $N_{h,\mathbf{u}}$ and $N_{h,p}$ the dimension of the one dimensional finite element space used along Ω_{1D} for $\mathbf{u}$ and p, respectively.

Figure 15.7 compares, at four different times, a reference Hi-Mod solution $\{\mathbf{u}^{R,h}_{m_{\mathbf{u}}}, p^{R,h}_{m_p}\}$, with $\mathbf{u}^{R,h}_{m_{\mathbf{u}}} = [u^{R,h}_{m_{\mathbf{u}},1}, u^{R,h}_{m_{\mathbf{u}},2}]^T$, computed by hierarchically reducing problem (15.12) with the Hi-POD solution $\{\mathbf{u}^h_{m_{\mathbf{u}}}(\boldsymbol{\alpha}^*), p^h_{m_p}(\boldsymbol{\alpha}^*)\}$, with $\mathbf{u}^h_{m_{\mathbf{u}}}(\boldsymbol{\alpha}^*) = [u^h_{m_{\mathbf{u}},1}(\boldsymbol{\alpha}^*), u^h_{m_{\mathbf{u}},2}(\boldsymbol{\alpha}^*)]^T$, for $l = 6$. The agreement between the two solutions is qualitatively very good, in spite of the fact that no information from the Hi-Mod solver on the problem after time t^5 is exploited to construct the Hi-POD solution. The pressure still features larger errors, as in the steady case.

We make this comparison more quantitative in Table 15.3, where we collect the $L^2(\Omega)$- and the $H^1(\Omega)$-norm of the relative error between the Hi-Mod reference solution and the Hi-POD one, at the same four times as in Fig. 15.7. We notice that the error does not grow significantly with time. This suggests that the Hi-POD approach can be particularly viable for reconstructing asymptotic solutions in periodic regimes, as in computational hemodynamics. As for the computational efficiency, Hi-POD solution requires $103s$ vs $287s$ of Hi-Mod one, with a significant reduction of the computational time.

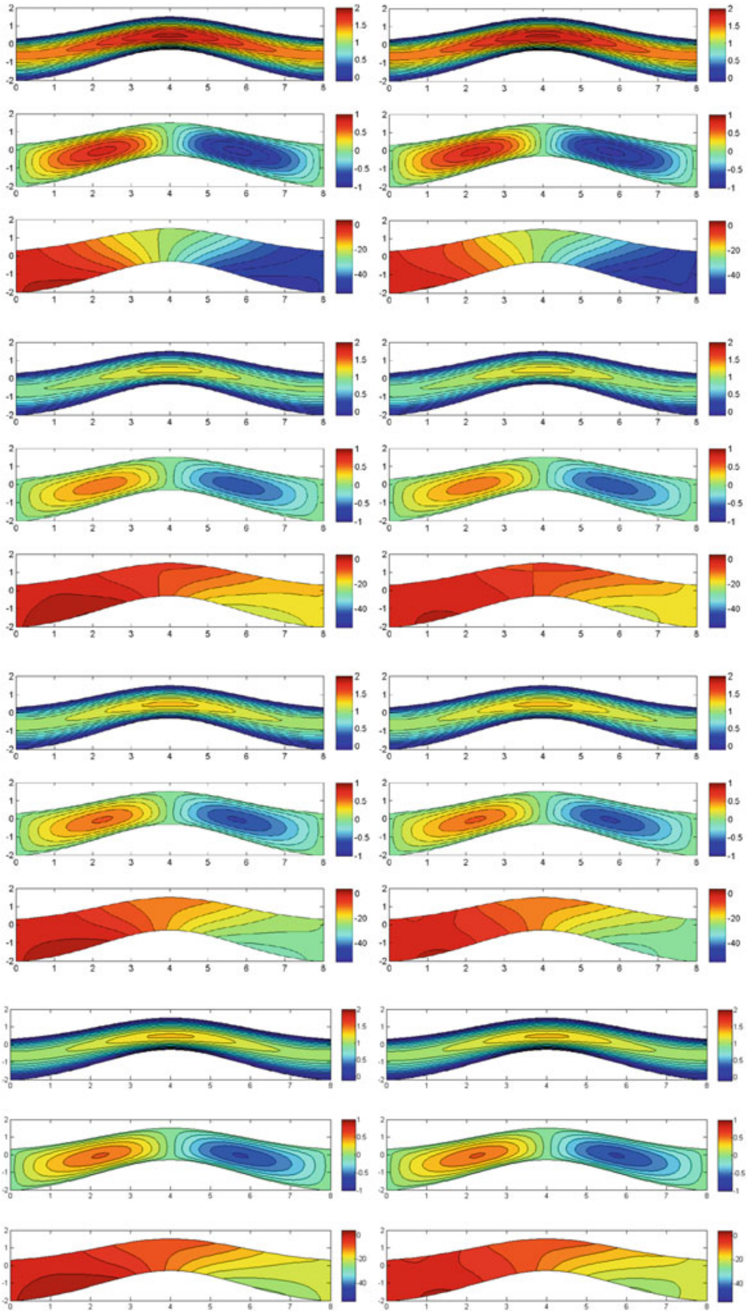


Fig. 15.7 Unsteady Navier-Stokes equations. Reference Hi-Mod solution (*left*) and Hi-Mod approximation yielded by the Hi-POD approach for $l = 6$ (*right*), at $t = 2$ (first row), $t = 4$ (second row), $t = 6$ (third row) and $t = T$ (fourth row): horizontal (*top*) and vertical (*middle*) velocity components; pressure (*bottom*)

Table 15.3 Unsteady Navier-Stokes equations

	$\dfrac{\|\|u^{R,h}_{m_{\mathbf{u}},1} - u^{h}_{m_{\mathbf{u}},1}(\boldsymbol{\alpha}^*)\|\|_{H^1(\Omega)}}{\|\|u^{R,h}_{m_{\mathbf{u}},1}\|\|_{H^1(\Omega)}}$	$\dfrac{\|\|u^{R,h}_{m_{\mathbf{u}},2} - u^{h}_{m_{\mathbf{u}},2}(\boldsymbol{\alpha}^*)\|\|_{H^1(\Omega)}}{\|\|u^{R,h}_{m_{\mathbf{u}},2}\|\|_{H^1(\Omega)}}$	$\dfrac{\|\|p^{R,h}_{m_p} - p^{h}_{m_p}(\boldsymbol{\alpha}^*)\|\|_{L^2(\Omega)}}{\|\|p^{R,h}_{m_p}\|\|_{L^2(\Omega)}}$
$t = 2$	$5.4 \cdot 10^{-4}$	$4.5 \cdot 10^{-4}$	$3.4 \cdot 10^{-2}$
$t = 4$	$2.4 \cdot 10^{-3}$	$2.1 \cdot 10^{-3}$	$1.0 \cdot 10^{-1}$
$t = 6$	$2.3 \cdot 10^{-3}$	$2.2 \cdot 10^{-3}$	$6.2 \cdot 10^{-2}$
$t = T$	$2.6 \cdot 10^{-3}$	$2.4 \cdot 10^{-3}$	$7.7 \cdot 10^{-2}$

Relative error associated with the Hi-Mod approximation provided by Hi-POD at different times

15.5 Conclusions and Future Developments

The preliminary results in Sects. 15.2.3, 15.3.1 and 15.4 yielded by the combination of the model/solution reduction techniques, Hi-Mod/POD, are very promising in view of modeling incompressible fluid dynamics in pipes or elongated domains. We have verified that Hi-POD enables a fast solution of parametrized ADR problems and of the incompressible, steady and unsteady, Navier-Stokes equations, even though in the presence of many (six) parameters. In particular, using Hi-Mod in place of a traditional discretization method applied to the reference (full) problem accelerates the offline phase and also the construction of the reduced problem projected onto the POD space.

Clearly, there are several features of this new approach that need to be investigated. First of all, we plan to migrate to 3D problems within a parallel implementation setting (in the library `LifeV`, www.lifev.org). Moreover, we aim at further accelerating the computational procedure by using empirical interpolation methods for possible nonlinear terms [24]. Finally, an extensive theoretical analysis is needed to estimate the convergence of the Hi-POD solution to the full one as well as the inf-sup compatibility of the Hi-Mod bases deserves to be rigorously analyzed.

As reference application we are interested in computational hemodynamics, in particular to estimate blood viscosity from velocity measures in patients affected by sickle cell diseases [23].

Acknowledgements The third and the fifth author gratefully acknowledge the NSF project DMS 1419060 “Hierarchical Model Reduction Techniques for Incompressible Fluid-Dynamics and Fluid-Structure Interaction Problems” (P.I. Alessandro Veneziani). The third author acknowledges also the financial support of GNCS-Project “Tecniche di riduzione della complessità computazionale per le scienze applicate”.

References

1. Aletti, M.: Educated bases for hierarchical model reduction in 2D and 3D. Master Thesis in Mathematical Engineering, Politecnico di Milano, Dec. 2013
2. Aletti, M., Perotto, S., Veneziani, A.: Educated bases for the HiMod reduction of advection-diffusion-reaction problems with general boundary conditions. MOX Report no **37/2015**.
3. Elman, H.C., Silvester, D.J., Wathen, A.J.: Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics. Numerical Mathematics and Scientific Computation, 2nd edn. Oxford University Press, Oxford, (2014)
4. Ern, A., Perotto, S., Veneziani, A.: Hierarchical model reduction for advection-diffusion-reaction problems. In: Kunisch, K., Of, G., Steinbach, O. (eds.) Numerical Mathematics and Advanced Applications, pp. 703–710. Springer, Heidelberg (2008)
5. Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th ed. Johns Hopkins University Press, Baltimore (2013)
6. Gunzburger, M.D.: Perspectives in Flow Control and Optimization. Advances in Design and Control, vol. 5. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (2003)
7. Guzzetti, S., Perotto, S., Veneziani, A.: Hierarchical model reduction for incompressible flows in cylindrical domains: the axisymmetric cae. Mox Report no S1/2016 (2016)
8. Hinze, M., Volkwein, S.: Error estimates for abstract linear-quadratic optimal control problems using proper orthogonal decomposition. Comput. Optim. Appl. **39**(3), 319–345 (2008)
9. Huanhuan, Y., Veneziani, A.: Efficient estimation of cardiac conductivities via POD-DEIM model order reduction. Appl. Numer. Math. **115**, 180–199 (2017)
10. Kahlbacher, M., Volkwein, S.: Galerkin proper orthogonal decomposition methods for parameter dependent elliptic system. Discuss. Math. Differ. Incl. Control Optim. **27**, 95–17 (2007)
11. Lions, J.-L., Magenes, E.: Non Homogeneous Boundary Value Problems and Applications. Springer, Berlin (1972)
12. Lupo Pasini, M.: HI-POD: HIerarchical Model Reduction Driven by a Proper Orthogonal Decomposition for Advection-Diffusion-Reaction Problems. Master Thesis in Mathematical Engineering, Politecnico di Milano, Dec. 2013
13. Lupo Pasini, M., Perotto, S., Veneziani, A. (2016, In preparation)
14. Mansilla Alvarez, L., Blanco, P., Bulant, C., Dari, E., Veneziani, A., Feijóo, R.: Transversally enriched pipe element method (TEPEM): an effective numerical approach for blood flow modeling. Int. J. Numer. Meth. Biomed. Eng. **33**, e2808 (2017). doi:10.1002/cnm.2808
15. Perotto., S.: Hierarchical model (Hi-Mod) reduction in non-rectilinear domains. In: Erhel, J., Gander, M., Halpern, L., Pichot, G., Sassi, T., Widlund, O. (eds.) Domain Decomposition Methods in Science and Engineering. Lecture Notes in Computational Science and Engineering, vol. 98, pp. 477–485. Springer, Cham (2014)
16. Perotto, S.: A survey of Hierarchical Model (Hi-Mod) reduction methods for elliptic problems. In: Idelsohn, S.R. (ed.) Numerical Simulations of Coupled Problems in Engineering. Computational Methods in Applied Sciences, vol. 33, pp. 217–241. Springer, Cham (2014)
17. Perotto, S., Veneziani, A.: Coupled model and grid adaptivity in hierarchical reduction of elliptic problems. J. Sci. Comput. **60**(3), 505–536 (2014)
18. Perotto, S., Zilio, A.: Hierarchical model reduction: three different approaches. In: Cangiani, A., Davidchack, R.L., Georgoulis, E., Gorban, A.N., Levesley, J., Tretyakov, M.V. (eds.) Numerical Mathematics and Advanced Applications, pp. 851–859. Springer, Berlin/Heidelberg (2013)
19. Perotto, S., Zilio, A.: Space-time adaptive hierarchical model reduction for parabolic equations. Adv. Model. Simul. Eng. Sci. **2**(25), 1–45 (2015)
20. Perotto, S., Ern, A., Veneziani, A.: Hierarchical local model reduction for elliptic problems: a domain decomposition approach. Multiscale Model. Simul. **8**(4), 1102–1127 (2010)
21. Perotto, S., Reali, A., Rusconi, P., Veneziani A.: HIGAMod: a Hierarchical IsoGeometric Approach for MODel reduction in curved pipes. Comput. Fluids **142**, 21–29 (2017)

22. Quarteroni, A., Saleri, F., Veneziani, A.: Factorization methods for the numerical approximation of Navier-Stokes equations. Comput. Methods Appl. Mech. Eng. **188**(1–3), 505–526 (2000)
23. Rivera, C.P., Veneziani, A., Ware, R.E., Platt, M.O.: Sickle cell anemia and pediatric strokes: computational fluid dynamics analysis in the middle cerebral artery. Exp. Biol. Med. (Maywood) **241**(7), 755–65 (2016). Epub 2016 Mar 4, doi:10.1177/1535370216636722
24. Rozza, G., Hesthaven, J.S., Stamm, B.: Certified Reduced Basis Methods for Parametrized Partial Differential Equations. SpringerBriefs in Mathematics. BCAM SpringerBriefs. Springer, Cham (2016)
25. Temam, R.: Navier-Stokes Equations. Theory and Numerical Analysis, third edition. North-Holland Publishing Co., Amsterdam (1984)
26. Veneziani, A., Viguerie, A.: Inexact algebraic factorization methods for the steady incompressible Navier-Stokes equations at moderate Reynolds numbers. TR-2017-002, Emory University (2016)
27. Veneziani, A., Villa, U.: ALADINS: an ALgebraic splitting time ADaptive solver for the Incompressible Navier-Stokes equations. J. Comput. Phys. **238**, 359–375 (2013)
28. Volkwein, S.: Proper Orthogonal Decomposition: Theory and Reduced-Order Modelling. Lecture Notes, University of Konstanz (2013)

Chapter 16
Adaptive Sampling for Nonlinear Dimensionality Reduction Based on Manifold Learning

Thomas Franz, Ralf Zimmermann, and Stefan Görtz

Abstract We make use of the non-intrusive dimensionality reduction method *Isomap* in order to emulate nonlinear parametric flow problems that are governed by the Reynolds-averaged Navier-Stokes equations. Isomap is a manifold learning approach that provides a low-dimensional embedding space that is approximately isometric to the manifold that is assumed to be formed by the high-fidelity Navier-Stokes flow solutions under smooth variations of the inflow conditions. The focus of the work at hand is the adaptive construction and refinement of the Isomap emulator: We exploit the non-Euclidean Isomap metric to detect and fill up gaps in the sampling in the embedding space. The performance of the proposed manifold filling method will be illustrated by numerical experiments, where we consider nonlinear parameter-dependent steady-state Navier-Stokes flows in the transonic regime.

16.1 Introduction

In [8], the authors proposed a non-intrusive low-order emulator model for nonlinear parametric flow problems governed by the Navier-Stokes equations. The approach is based on the manifold learning method Isomap [17] combined with an interpolation scheme and will be referred to hereafter as Isomap+I. Via this method, a low-dimensional embedding space is constructed that is approximately isometric to the manifold that is assumed to be formed by the high-fidelity Navier-Stokes flow

T. Franz (✉) • S. Görtz
Institute for Aerodynamics and Flow Technology, German Aerospace Center (DLR), Braunschweig, Germany
e-mail: thomas.franz@dlr.de; stefan.goertz@dlr.de

R. Zimmermann
Institute 'Computational Mathematics', TU Braunschweig, Braunschweig, Germany

Department of Mathematics and Computer Science, University of Southern Denmark, Odense M, Denmark
e-mail: ralf.zimmermann@tu-bs.de; zimmermann@imada.sdu.dk

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_16

solutions under smooth variations of the inflow conditions. As with almost all model reduction methods, the offline stage for the Isomap+I approach requires a suitable design of experiment, i.e., a well-chosen sampling of high-fidelity flow solutions, the so-called snapshots. The online stage, however, might be considered as an adaptive way for choosing for each low-order prediction the most suitable local snapshot neighborhood rather than using all available snapshot information in a brute-force way. The notion of locality is based on the Isomap metric. The focus of this article is on an adaptive construction and refinement of the underlying design of experiment. Since Isomap comes with a natural non-Euclidean metric for measuring snapshot distances, we make use of this metric to detect gaps in the embedding space. By the (approximate) isometry between the embedding space and the manifold of flow solutions, we obtain in this way a *manifold filling* design of experiment. In contrast, standard approaches like the Latin Hypercube method [6] aim at a *parameter-space filling* design of experiment. The performance of the proposed manifold filling method is illustrated by numerical experiment, where we consider nonlinear parameter-dependent steady-state Navier-Stokes flows in the transonic regime.

Organization In Sect. 16.2, the Isomap-based emulator model is briefly introduced. The adaptive sampling strategy based on the manifold characterization is developed in Sect. 16.3.1, followed by a proof of concept in Sect. 16.3.2. Afterwards, the methods are demonstrated for an engineering application in Sect. 16.4. Finally, conclusions are drawn in Sect. 16.5.

16.2 The Isomap-Based Emulator Model

In this section, we briefly review the manifold learning based approach to emulate steady-state flows governed by the Reynolds-averaged Navier Stokes (RANS) equations that was introduced in [7, 8]. For background information on computational fluid dynamics see, e.g., [3], for an introduction to differentiable manifolds see, e.g., [16].

Let $\mathscr{M} \subset \mathbb{R}^n$ be an embedded submanifold in the Euclidean space with intrinsic dimension $\dim(\mathscr{M}) = d < n$. Let $\mathscr{W} \subset \mathscr{M}$ be an open domain in $\mathscr{M}$ such that there exists a coordinate chart[1] $h : \mathscr{W} \to \mathscr{Y}$ onto an open domain $\mathscr{Y} \subset \mathbb{R}^d$. The fundamental objective of manifold learning (ML) [5, 18] is to solve the *isometric embedding problem* [2, 18], which we reformulate as follows:

> For a given finite set of sampled data points $W = \{\mathbf{W}^1, \dots, \mathbf{W}^m\} \subset \mathscr{W} \subset \mathbb{R}^n$ compute an approximation of the coordinate chart h such that the restriction to the discrete sample points
>
> $$h|_W : \mathscr{W} \supset W = \{\mathbf{W}^1, \dots, \mathbf{W}^m\} \to Y = \{\mathbf{y}^1, \dots, \mathbf{y}^m\} \subset \mathscr{Y}, \quad h(\mathbf{W}^i) = \mathbf{y}^i,$$

[1] i.e., a bijective both-ways differentiable mapping.

is such that the image point set Y features (approximately) the same inter-point distances as the high dimensional data set W.

One of the most popular ML methods is *Isomap* [17]. Isomap works by approximating the *geodesic distance* between data vectors $\mathbf{W}^i$ and $\mathbf{W}^j$ via computing the length of an Euclidean polygon course that connects $\mathbf{W}^i$ and $\mathbf{W}^j$. The polygon course is determined based on a graph-theoretical shortest path problem, which is detailed in [8] and [17]. The basic idea is illustrated in Fig. 16.1a. Once the geodesic distances are estimated, a distance matrix $D \in \mathbb{R}^{m \times m}$ is formed, where the entry d_{ij}, $i, j = 1, \ldots, m$, is the approximated geodesic distance between $\mathbf{W}^i$ and $\mathbf{W}^j$. The next step is to employ classical multidimensional scaling [11, Sect. 14] with the distance matrix D as an input. This results in a data set $Y = \{\mathbf{y}^1, \ldots, \mathbf{y}^m\}$ with $\left\|\mathbf{y}^i - \mathbf{y}^j\right\| \approx d_{ij}$ for $i, j = 1, \ldots, m$. Moreover, the data set Y is tuned for the envisioned application by minimizing an additional *loss function* afterwards, see [7, Sect. 4.3.1]. The resulting embedding space when applying Isomap to the 'swiss roll' standard example in manifold learning (see Fig. 16.1b) is displayed in Fig. 16.1c.

So far, we have constructed a low-dimensional representation of the high-dimensional input data. In order to obtain a valid emulator, a mapping from the low-dimensional space to the high-dimensional manifold is required. As it is common in many model reduction methods, including proper orthogonal decom-

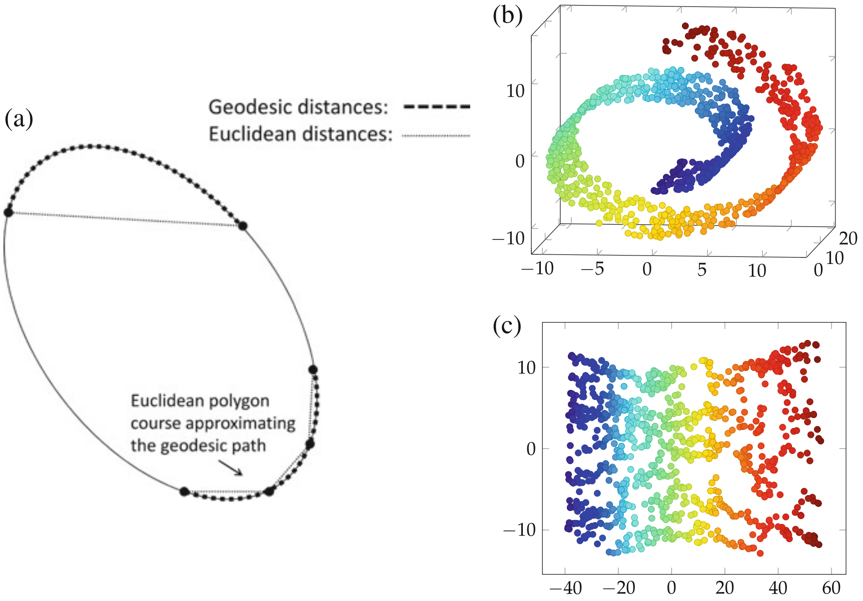


Fig. 16.1 *Left*: Geodesic distances vs. Euclidean distances. *Right*: The 'swiss roll' standard example. (**a**) Approximation of the geodesic distance. (**b**) Swiss roll: Original data set in $\mathbb{R}^3$. (**c**) Swiss roll: Isometric embedding in $\mathbb{R}^2$

position [12] and the reduced basis method [15], we assume that the output of the emulator is a linear combination of the input snapshots. In our setting, the input data vectors stem from solutions to the RANS equations under parametric variations, i.e., $\mathbf{W}^j = \mathbf{W}(\mathbf{p}^j)$, where $\mathbf{p}^j$ is the parameter vector specifying the inflow conditions. The ansatz at an untried flow condition $\mathbf{p}^*$ is $\mathbf{W}(\mathbf{p}^*) = \sum_{j=1}^m a_j(\mathbf{p}^*)\mathbf{W}^j$. Hence, the nonlinear parametric dependency is in the coefficients $a_j = a_j(\mathbf{p})$ while the snapshots $\mathbf{W}^j$ are fixed. The essential idea of manifold learning is to localize the information in the sense that only the N nearest neighbors $\{\mathbf{W}^j = \mathbf{W}(\mathbf{p}^j) | j \in \mathscr{I}, |\mathscr{I}| = N\}$ contribute to $\mathbf{W}(\mathbf{p}^*)$, where the notion of proximity depends on the Isomap metric.

The exact procedure is as follows: If the flow at $\mathbf{p}^*$ is to be emulated, we then first determine the corresponding location in the embedding space $\mathbf{y}^* = \mathbf{y}(\mathbf{p}^*) \in \mathbb{R}^d$ via multivariate interpolation based on the embedded data set $\{(\mathbf{p}^j, \mathbf{y}^j)\}_{j=1}^m$. Isomap provides us with the nearest neighbors $\{\mathbf{y}^j \mid j \in \mathscr{I}\}$ of $\mathbf{y}^*$. Next, we represent $\mathbf{y}^*$ approximatively as a weighted linear combination of the nearest neighbors as $\mathbf{y}^* \approx \sum_{j\in\mathscr{I}} a_j \mathbf{y}^j$, where we determine the weights a_j via the following optimization problem:

$$\min_{\mathbf{a}\in\mathbb{R}^N} \|\mathbf{y}^* - \sum_{j\in\mathscr{I}} a_j \mathbf{y}^j\|_2^2 + \|\mathbf{a}\|_c^2 \qquad \text{s. t.} \quad \sum_{j\in\mathscr{I}} a_j = 1, \tag{16.1}$$

with penalty term

$$\|\mathbf{a}\|_c^2 := \sum_{j\in\mathscr{I}} c_j a_j^2, \quad c_j = \varepsilon \left(\frac{\|\mathbf{y}^* - \mathbf{y}^j\|_2}{\max_i\{\|\mathbf{y}^* - \mathbf{y}^i\|_2\}} \right)^k, \quad 0 < \varepsilon \ll 1,\ 1 < k \in \mathbb{N}.$$

The penalty term weights the influence of the snapshots based on their distance to the prediction point $\mathbf{y}^*$. Let $\mathbf{a}^* \in \mathbb{R}^N$ be the solution to (16.1). Because of the inherent (approximate) isometry between the snapshots $\mathbf{W}^j$ and the locations $\mathbf{y}^j$ in the embedding space, we use the same weight vector to construct the high-dimensional flow state

$$\mathbf{W}^* = \sum_{j\in\mathscr{I}} a_j^* \mathbf{W}^j. \tag{16.2}$$

The extra condition in Eq. (16.1) is such that when the whole set of embedded snapshots $y^j, j \in \mathscr{I}$, is translated via $T : y \mapsto y + \mu$ to a new set $z^j = T(y_j)$, $j \in \mathscr{I}$, then

$$T(y^*) = y^* + \mu = (y^1, \ldots, y^{|\mathscr{I}|})a + \mu = (z^1, \ldots, z^{|\mathscr{I}|})a = z^*.$$

Best practice settings for the meta-parameters ε, k and further details are given in [7]. In addition, a heuristic choosing the size of the neighborhood $\mathscr{I}$ automatically is developed in [7] and employed for all conducted predictions. We call the above process *Isomap+I.*

16.3 Adaptive Sampling

The algorithmic efficiency and the numerical accuracy of the Isomap-based emulator strongly depend on the selected input information. Computing the input snapshots is costly by nature, because high-fidelity solutions to the very system that is to be emulated are required. Moreover, spatial sampling methods suffer from the *curse of dimensionality* [6, Sect. 1.1] in the sense that the number of sample points that is required to achieve a certain sampling density grows exponentially with the spatial dimension. To keep the number of full system solves as small as possible, we present an incremental sampling method that attempts to create a homogeneously distributed data set of the manifold based on geometric information.

16.3.1 Manifold Filling Adaptive Sampling Strategies

As outlined in Sect. 16.2, Isomap preserves the interpoint distances of the underlying manifold domain $\mathscr{W}$. This property is what we exploit for detecting gaps in the input data set.

Let $\{\mathbf{y}^1, \dots, \mathbf{y}^m\} = Y_m \subset \mathbb{R}^d$ be the low-dimensional representative of the large-scale input snapshot set $\{\mathbf{W}^1, \dots, \mathbf{W}^m\} = W_m \subset \mathbb{R}^n$ and let $\mathbf{y} : \mathscr{P} \to \mathbb{R}^d$ with $\mathbf{y}(\mathbf{p}^j) = \mathbf{y}^j$, $\mathbf{p}^j \in \mathscr{P} \subset \mathbb{R}^d, j = 1, \dots, m$. If there is a location $\mathbf{y}_g \in \{\mathbf{y}(\mathbf{p}) \mid \mathbf{p} \in \mathscr{P}\}$ and a radius $\gamma > 0$ such that the γ-ball $B_\gamma(\mathbf{y}_g) = \{\tilde{\mathbf{y}} \in \mathbb{R}^d \mid \|\mathbf{y}_g - \tilde{\mathbf{y}}\|_2 < \gamma\}$ does not contain any sampled representatives, i.e., $\mathbf{y}^j \notin B_\gamma(\mathbf{y}_g) \forall j = 1, \dots, m$, then we say that there is a *gap* of size γ at $\mathbf{y}_g \in \{\mathbf{y}(\mathbf{p}) \mid \mathbf{p} \in \mathscr{P}\}$. The objective is to detect these gaps and fill them by adding suitable snapshots to the input data set.

We device an iterative adaptation process. Let $\mathscr{P} \subset \mathbb{R}^d$ be the parameter domain of interest and let $P_{\tilde{m}} = \{\mathbf{p}^1, \dots, \mathbf{p}^{\tilde{m}}\} \subset \mathscr{P}$ be a set of $\tilde{m} \in \mathbb{N}$ preselected sample locations. Moreover, let $1 \leq i \leq m - \tilde{m}$ be the number of the current iteration of the adaptive sampling process, where $i, m \in \mathbb{N}$ and $m > \tilde{m}$ is the maximal number of affordable snapshots. Starting with the initial design of experiment (DoE) of $\tilde{m}$ snapshots $W_{\tilde{m}} = \{\mathbf{W}^1, \dots, \mathbf{W}^{\tilde{m}}\} \subset \mathbb{R}^n$, where $\mathbf{W}^j = \mathbf{W}(\mathbf{p}^j)$, the associated initial embedding $Y_{\tilde{m}} = \{\mathbf{y}^1, \dots, \mathbf{y}^{\tilde{m}}\} \subset \mathbb{R}^d$ is calculated via Isomap.

The procedure to detect gaps is as follows: For a given location $\mathbf{p} \in \mathscr{P}$ the corresponding location in the embedding space $\mathbf{y} : \mathscr{P} \to \mathbb{R}^d$ is determined via interpolation based on the data set of current sample locations $\{(\mathbf{p}^j, \mathbf{y}^j)\}_{j=1}^{\tilde{m}}$, cf. Sect. 16.2. Then, the *weighted sum of the distances of the* $N \in \mathbb{N}$ *nearest neighbors* $y^j, j \in \mathscr{I}$ to $\mathbf{y}(\mathbf{p})$ is calculated:

$$dist(\mathbf{y}(\mathbf{p})) := \frac{d_{min}(\mathbf{y}(\mathbf{p}))}{d_{max}(\mathbf{y}(\mathbf{p}))} \sum_{j \in \mathscr{I}} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}^j\|_2, \qquad (16.3)$$

Algorithm 1 Manifold filling adaptive sampling algorithm

```
Require: Desired number of snapshots $m$, number of initial snapshots $\tilde{m}$
Generate $\tilde{m} < m$ parameter values $\mathbf{p}^1, \dots, \mathbf{p}^{\tilde{m}} \in \mathscr{P}$, e.g. via LHS
$P \leftarrow \{\mathbf{p}^1, \dots, \mathbf{p}^{\tilde{m}}\}$
Compute snapshot solutions $\mathbf{W}(\mathbf{p})$ at each parameter value $\mathbf{p} \in P$
$W \leftarrow \{\mathbf{W}(\mathbf{p}^1), \dots, \mathbf{W}(\mathbf{p}^{\tilde{m}})\}$                ▷ initial sampling
for $i = 1$ to $m - \tilde{m}$ do
    Calculate embedding $Y$ of the generated snapshot set $W$ via Isomap
    Compute interpolation model for $\mathbf{y}$ based on $\{(\mathbf{p}^j, \mathbf{y}^j)\}_{j=1}^{\tilde{m}+i-1}$
    Determine $\mathbf{p}^* \in \mathscr{P}$ by maximizing $E_{dist}$ or $E_{rec}$
    Compute snapshot solution $\mathbf{W}^*$ at parameter configuration $\mathbf{p}^* \in \mathscr{P}$
    $P \leftarrow P \cup \{\mathbf{p}^*\}$
    $W \leftarrow W \cup \{\mathbf{W}^*\}$
end for
return Set $W$ of $m$ snapshots
```

where $d_{min}(\mathbf{y}(\mathbf{p})) = \min_{j\in\mathscr{I}} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}^j\|_2$ and $d_{max}(\mathbf{y}(\mathbf{p})) = \max_{j\in\mathscr{I}} \|\mathbf{y}(\mathbf{p}) - \mathbf{y}^j\|_2$. The distance function (16.3) is multiplied by an indicator function ω:

$$E_{dist}(\mathbf{y}(\mathbf{p})) := dist(\mathbf{y}(\mathbf{p})) \cdot \omega(\mathbf{p}), \quad \omega(\mathbf{p}) = \begin{cases} 1 & \text{if } \mathbf{p} \in \mathscr{P}, \\ 0 & \text{else,} \end{cases} \tag{16.4}$$

which ensures that the adaptation process takes place only in the inside of the parameter domain of interest. The maximizer $\mathbf{p}^* = \arg\max E_{dist}(\mathbf{y}(\mathbf{p}))$ determines the next snapshot to be added to the model. The above method will be referred to as the maximum distance error (MDE) strategy. A pseudo code of this method is outlined in Algorithm 1.

On top of the distance based error criterion (16.4), we introduce a reconstruction error indicator that yields reliable results when the manifold is sufficiently homogeneously sampled, i.e. the sampling does not divide into disconnected clusters. Let $Y_{\tilde{m}+i-1}$ be the embedding data set at iteration $i-1$ of the adaptive sampling process. For each $\mathbf{y}^j \in Y_{\tilde{m}+i-1}$, we compute a prediction $\hat{\mathbf{W}}(\mathbf{y}^j) = \hat{\mathbf{W}}^j$ based on its N nearest neighbors and the relative error $E_{rel}(\mathbf{y}^j) = \frac{\|\hat{\mathbf{W}}^j - \mathbf{W}^j\|_2}{\|\mathbf{W}^j\|_2}$ to the corresponding snapshot $\mathbf{W}^j$. Note that $\mathbf{y}^j$ is not counted as a neighbor of itself and hence $\hat{\mathbf{W}}^j \neq \mathbf{W}^j$. Subsequently, interpolation is performed to approximate the relative error at an arbitrary location $\mathbf{y} \notin Y_{\tilde{m}+i-1}$ based on the data set $\{(\mathbf{y}^j, E_{rel}(\mathbf{y}^j))\}_{j=1}^{\tilde{m}+i-1}$. To ensure that the error is zero at the given sample points, the reconstruction error is defined as

$$E_{rec}(\mathbf{y}(\mathbf{p})) := E_{rel}(\mathbf{y}(\mathbf{p})) \cdot E_{dist}(\mathbf{y}(\mathbf{p})). \tag{16.5}$$

Since an almost homogeneously sampled manifold must be given, we employ the error function E_{rec} only every kth iteration in practice. For the remaining iterations

E_{dist} is utilized exclusively to ensure a homogeneously distributed manifold. The resulting hybrid error sampling strategy is referred to as HYE in the following.

Remark 1 It is not a necessity that we add only one snapshot per iteration. In each iteration, we may choose to determine several local maximizers to E_{dist} and E_{rec}, respectively, and add the corresponding snapshots to the information pool.

Choice of the Initial Sampling Plan and Starting Points for Optimization When starting from scratch, the initial sampling plan of $\tilde{m}$ points in the parameter domain of interest $\mathscr{P}$ is chosen randomly. More precisely, we employ either space filling random Latin Hypercube Sampling (LHS) [6] or Halton sequences [9] to construct the initial DoE. The selection of the starting points for the maximization of either (16.4) or (16.5) requires special consideration as the objective functions features many local maxima. We make the following differentiation:

1. If the initial DoE $P = \{\mathbf{p}^1, \dots, \mathbf{p}^{\tilde{m}}\}$ is such that its convex hull coincides with the parameter space $\mathscr{P}$ of interest, then we treat the convex hull of the corresponding embedding points $Y = \{\mathbf{y}^1, \dots, \mathbf{y}^{\tilde{m}}\}$ as the domain of interest in the embedding space, even though the mapping is not convex in general. In this case, we perform a *Delaunay triangulation* [14] of Y and determine the centers $\mathbf{y}(\mathbf{c}^i) \in \text{conv}(Y)$, $i = 1, \dots, l$ of the Delaunay simplices of largest volume. The corresponding locations $\mathbf{p}(\mathbf{y}(\mathbf{c}^i)) \in \mathscr{P}$ are selected as starting points for optimizing (16.4). (The $\mathbf{p}(\mathbf{y}(c^i))$ are found via interpolation.)
2. Otherwise, we perform another space filling LHS to create the starting points randomly in order to avoid clustering effects. This procedure is also followed for determining the starting points for optimizing (16.5) in order to increase the probability to locate the global maximum.

16.3.2 Proof of Concept

In this section, we illustrate the performance of Algorithm 1 on two academic examples.

Detection of Gaps Reconsider the swiss role, parameterized by two parameters t and h:

$$\mathbf{s} : \mathscr{P} \to \mathscr{W} \subset \mathbb{R}^3, \quad (t, h) \mapsto (t\cos(t), h, t\sin(t)), \quad \mathscr{P} = [\tfrac{3}{2}\pi, \tfrac{9}{2}\pi) \times [0, 21)$$

To artificially create a hole in the sample set, we exclude the rectangle $(9.5, 10.5) \times (8, 13)$ from the parameter domain and construct an initial random-based DoE P of $|P| = 748$ sample points in $\mathscr{P} \setminus (9.5, 10.5) \times (8, 13)$.

Now, we conduct a single step of Algorithm 1, where we perform step 8 with respect to (16.4) and consider only the single nearest neighbor in evaluating the distance function (16.3). This results in an optimal location $\mathbf{p}^* \in \mathscr{P}$ that is displayed

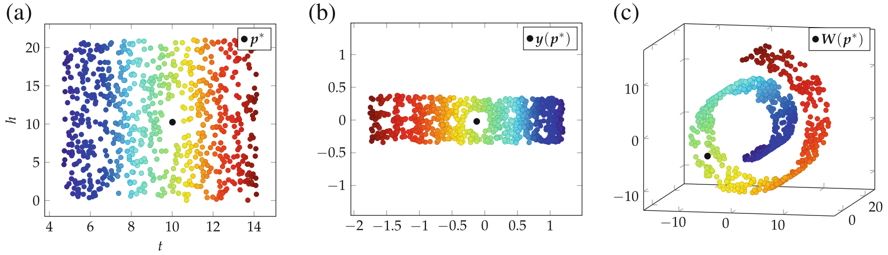


Fig. 16.2 Detection of a gap in the DoE illustrated for the swiss roll. (**a**) Point cloud and detected gap center in the parameter domain $\mathscr{P}$. (**b**) Point cloud and detected gap center in the embedding domain $\mathscr{Y}$. (**c**) Point cloud and detected gap center on the swiss roll manifold $\mathscr{W}$

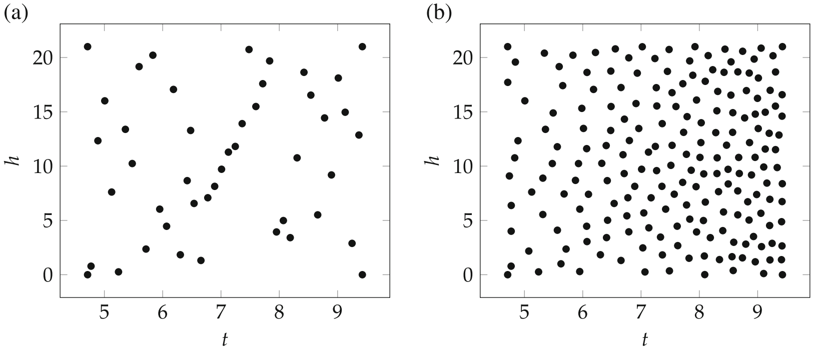


Fig. 16.3 Curved plate: Locations of the initial and refined parameter samples. (**a**) Initial parameter sampling. (**b**) Refined sample set

in Fig. 16.2a. Figure 16.2b, c depict the corresponding point $\mathbf{y}^* = \mathbf{y}(\mathbf{p}^*)$ in the embedding space and $\mathbf{s}(\mathbf{p}^*) \in \mathbb{R}^3$ on the swiss roll manifold, respectively.

Manifold Filling As a second academic example, we consider a curved plate parameterized by

$$\mathbf{c} : \mathscr{P} \to \mathscr{W} \subset \mathbb{R}^3, (t,h) \mapsto (\tfrac{t^2}{10}\cos(t), h, \tfrac{t^2}{10}\sin(t)), \quad \mathscr{P} = [\tfrac{3}{2}\pi, 3\pi] \times [0, 21].$$

We start with a Latin hypercube sampling of 40 data points selected from the interior of $\mathscr{P}$ and add the four corner points of the rectangle $\mathscr{P}$, see Fig. 16.3a. The corresponding initial sample data set $W_{44} \subset \mathscr{W}$ and its discrete Isomap embedding Y_{44} are depicted in Fig. 16.4a, b, respectively.

We detect the regions of low sampling density via the MDE approach. The starting points for the optimization procedures are chosen by a LHS of size 30 in each iteration. In Fig. 16.4c, d, the generated snapshot set $W_{\tilde{m}+i}$ and its embeddings

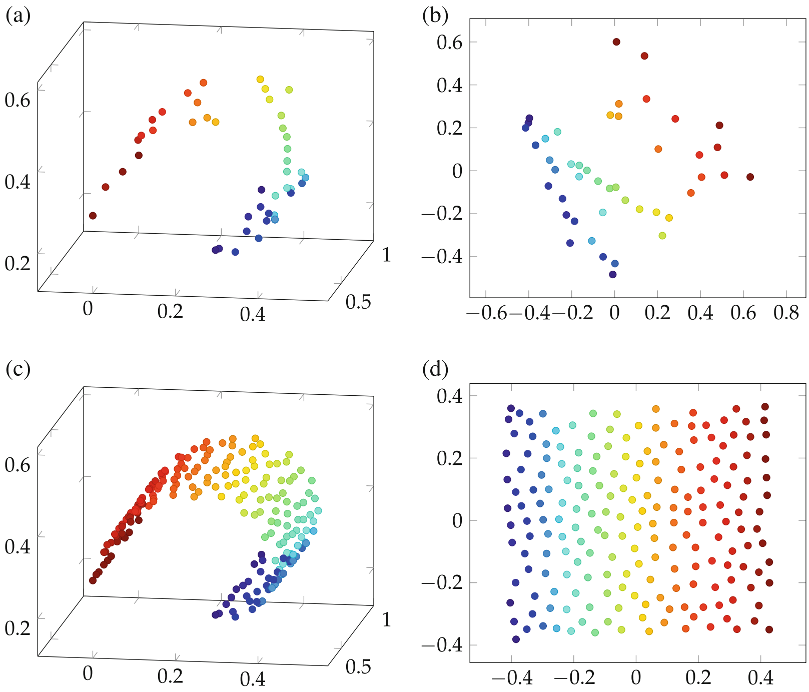


Fig. 16.4 Manifold filling adaptive sampling strategy illustrated for a curved plate. (**a**) $W_{44} \subset \mathscr{W}$. (**b**) $Y_{44} \subset \mathscr{Y}$. (**c**) $W_{194} \subset \mathscr{W}$. (**d**) $Y_{194} \subset \mathscr{Y}$

after $i = 150$ iterations is shown, respectively.[2] The $\tilde{m}+i = 194$ parameter locations in $\mathscr{P}$ associated with the final refined snapshot set are depicted in Fig. 16.3b. Note that the sampling plan is denser for larger t, which is in line with the fact that the function $\mathbf{c}$ exhibits a higher angular velocity for increasing t.

16.4 An Engineering Example

As an engineering application, we emulate the high-Reynolds number flow past the two-dimensional NACA 64A010 airfoil in the transonic flow regime. The geometry of the airfoil is shown in Fig. 16.5b. The hybrid unstructured grid features $21,454$ grid points, including 400 surface grid points, and is depicted in Fig. 16.5.

The objective is to emulate the distribution of the pressure coefficient C_p on the surface of the airfoil under varying angle of attack, α, and Mach number, Ma. To

[2] The number of nearest neighbors used for the embedding was chosen automatically in each iteration according to [7, Sect. 4.3.3].

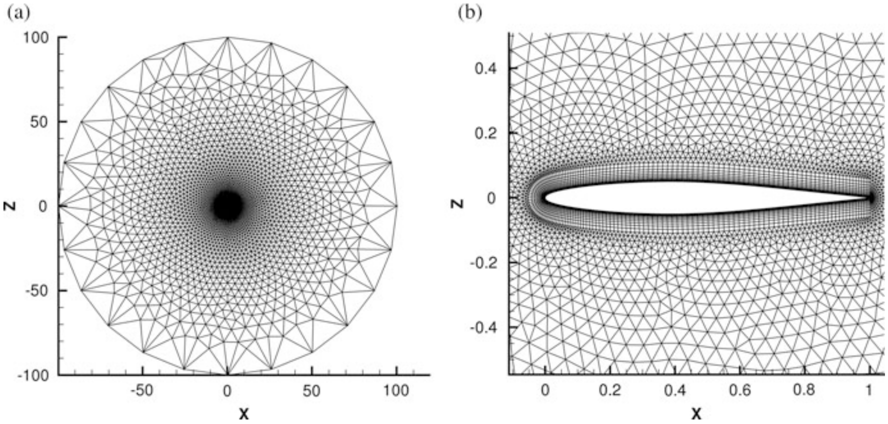


Fig. 16.5 Computational grid for the NACA 64A010 airfoil. (**a**) View of the entire flow field. (**b**) Detailed view close to the surface

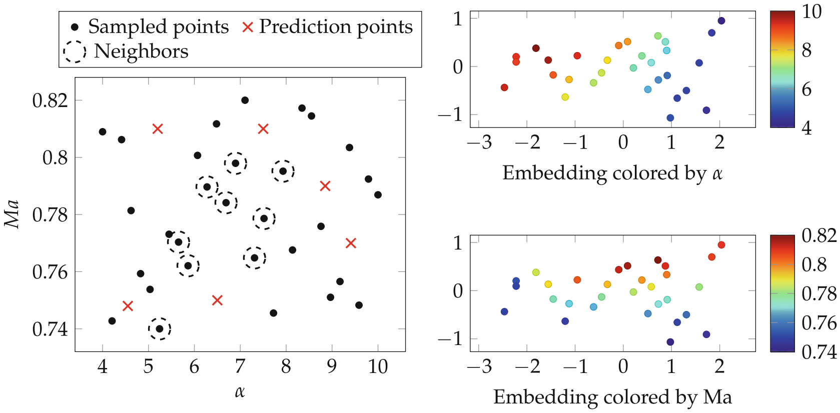


Fig. 16.6 *Left*: Locations of the snapshots and various prediction points in the α-Ma-space for the NACA 64A010 test case. Furthermore, the employed snapshots for the prediction at $(\alpha, \mathrm{Ma}) = (6.5°, 0.75)$ are encircled. *Right*: Representatives within the embedding space colored corresponding to the angle of attack α (*top*) and the Mach number Ma (*bottom*)

this end, we generate a snapshot set of flow solutions, where the initial parameter locations P are selected via a LHS of $m = 30$ samples from in the parameter space $\mathscr{P} = \{(\alpha, \mathrm{Ma}) \in [4°, 10°] \times [0.74, 0.82]\}$, see Fig. 16.6. The corresponding viscous flow solution snapshots $\mathbf{W}(\mathbf{p})$, $\mathbf{p} \in P$, are computed with DLR's RANS solver TAU [10] using the negative Spalart-Allmaras one-equation turbulence model [1]. Convergence is detected based on a reduction of the normalized density residual by seven orders of magnitude in each solver run. The Reynolds number is fixed throughout at a value of Re $= 7{,}500{,}000$. Computing a full CFD solution under this conditions took 474 iterations or 63 CPU seconds on average.[3]

From the flow solution snapshots, we extract the vectors $\mathbf{W}(\mathbf{p})$, $\mathbf{p} \in P$ containing the discretized surface-C_p distributions, which form our initial point cloud W. Since in this test case two varying parameters are considered, the full-order solution manifold $\mathscr{W} = \{\mathbf{W}(\alpha, \mathrm{Ma}), \quad (\alpha, \mathrm{Ma}) \in \mathscr{P}\} \subset \mathbb{R}^{400}$ is of intrinsic dimension two.[4] The low intrinsic dimension is not a technical requirement, but an natural assumption in the context of model order reduction. We use the Isomap+I process of Sect. 16.2 to predict the C_p distributions at untried parameter locations and compare the results to the approximations computed via proper orthogonal decomposition combined with interpolation, which yields predictions at untried parameter combinations by interpolating the POD coefficients as done in [4]. This method will be referred to as POD+I in the following. Both interpolation based ROMs are coupled with the RBF interpolation using a TPS kernel augmented by a polynomial $\varphi \in \Pi_1$ [6, 13], $\varphi : \mathbb{R}^d \to \mathbb{R}$, where Π_1 is the space of polynomials of degree of at most one. Prior to each interpolation process, the sample locations in the parameter space are scaled to the unit hypercube, with the result that the input scaling is normalized and does not thwart the Isomap metric. The TPS kernel has been chosen for its good approximation quality and robustness based on best practice observations made in [19]. The first author's thesis features the results at all the prediction points indicated in Fig. 16.6. Here, we display only the worst result, which is obtained at $(\alpha, \mathrm{Ma}) = (6.5°, 0.75)$, since we aim at improving the prediction by adaptively refining the snapshot sampling according to the MDE and HYE strategy. The nine nearest neighbors on the manifold that are used to compute the prediction are encircled in Fig. 16.6 and the resulting C_p distribution is shown in Fig. 16.7.

We start with an initial DoE of 5 sample points generated by a Halton sequence, where none of the points is considered to lie on the boundary of $\mathscr{P}$. We perform 25 iterations of Algorithm 1 to arrive at a final sampling of 30 snapshots. In both sampling strategies, we consider only the nearest neighbor when evaluating the objective function (16.4). In the hybrid strategy HYE, we maximize (16.5) instead of (16.4) in every third iteration. In Table 16.1, we list the mean relative error, the standard deviation and the maximum relative error for the Isomap emulator associated with the adaptively refined data sets obtained via the MDE strategy and the HYE strategy, respectively.[5] For comparison, we include the errors corresponding to Isomap emulators based on the non-adaptive random DoEs of the same cardinality 30 that are obtained by a Halton sequence and a space filling

[3] All computations were conducted sequentially on the same standard desktop computer endowed with an Intel® Xeon® E3-1270 v3 Processor (8M Cache, 3.50 GHz) and 32 GB RAM.

[4] For applications where the dimension of the manifold is unknown, there exist various methods to estimate the intrinsic dimensionality of the data, e.g. by looking for the "elbow" [17].

[5] Error quantification is with respect to the surface C_p distributions and is based on 2500 uniformly distributed TAU reference CFD solutions.

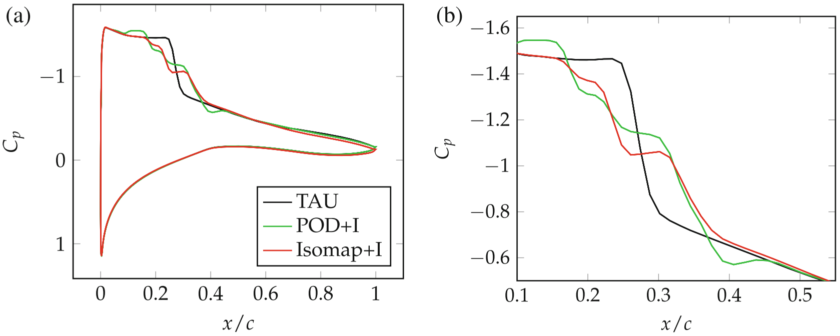


Fig. 16.7 Surface C_p distribution at $(\alpha, \mathrm{Ma}) = (6.5°, 0.75)$. The upper and lower curves correspond to the suction and pressure side of the airfoil, respectively. Results obtained based on a non-adaptive random sampling of 30 input snapshots. (**a**) Complete surface C_p distribution. (**b**) Detailed view close to shock

Table 16.1 The mean relative error, its standard deviation and the maximum relative error after a full sampling process of various sampling strategies/designs for the NACA 64A010 test case

Method	Mean rel. error	STD. deviation	Max. rel. error
MDE	$2.3347 \cdot 10^{-2}$	$1.6616 \cdot 10^{-2}$	$9.2956 \cdot 10^{-2}$
HYE	$2.1903 \cdot 10^{-2}$	$1.0320 \cdot 10^{-2}$	$5.3337 \cdot 10^{-2}$
Halton	$2.6670 \cdot 10^{-2}$	$2.7398 \cdot 10^{-2}$	$2.3016 \cdot 10^{-1}$
LHS	$3.1262 \cdot 10^{-2}$	$2.6257 \cdot 10^{-2}$	$1.8009 \cdot 10^{-1}$

LHS. The adaptive sampling strategies developed here yield samplings with a smaller change of the relative errors than in both random samplings. Hence the maximum relative error is closer to the mean relative error, which leads to a more reliable global emulator with less outliers in prediction accuracy. Note, that the mean relative errors are also smaller for the adaptive strategies. The embeddings of the final samplings are shown in Fig. 16.8. As aspired by MDE, the embedding of the corresponding sampling is quite evenly distributed. This also holds for the embedding of the sampling obtained by HYE, even if E_{rec} is applied in every third iteration. In contrast, the embeddings of both random samplings feature close-by points, which may lead to redundant information.

We use the HYE-adaptively constructed emulator to predict the surface pressure at the flow condition of $(\alpha, \mathrm{Ma}) = (6.5°, 0.75)$, where a poor approximation quality was observed in Fig. 16.7. Recall that those results were obtained with the same number of 30 input snapshots, but chosen randomly (LHS) rather than adaptively.

The C_p-distributions obtained from the emulators are shown in Fig. 16.9a, where we compare the CFD reference and the Isomap+I and the POD+I emulators. As can be seen, both the Isomap+I and the POD+I predictions greatly benefit from the adaptive sampling process. (Compare Figs. 16.7, 16.8, and 16.9a.) The Isomap+I prediction matches the reference solution with high accuracy throughout by using

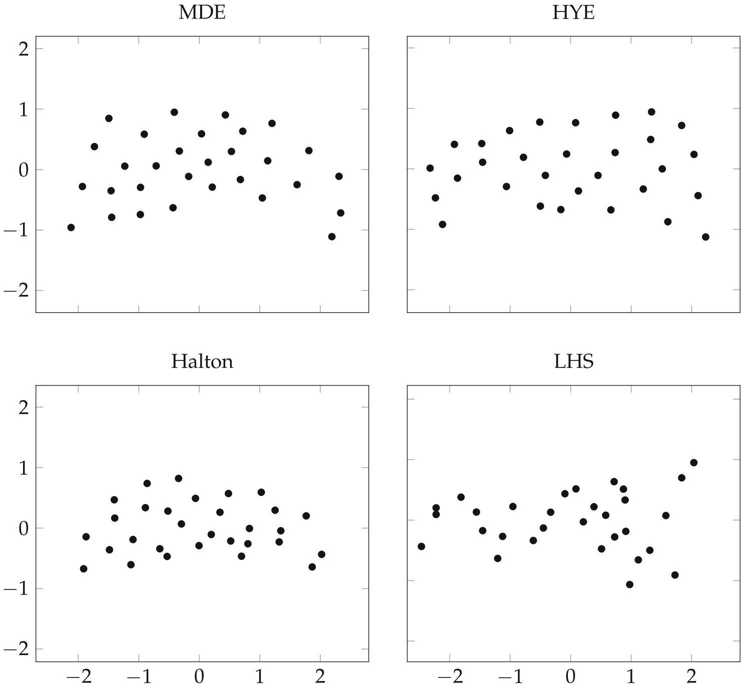


Fig. 16.8 Embeddings of the final samplings obtained by various sampling methods and DoEs for the NACA 64A010 test case

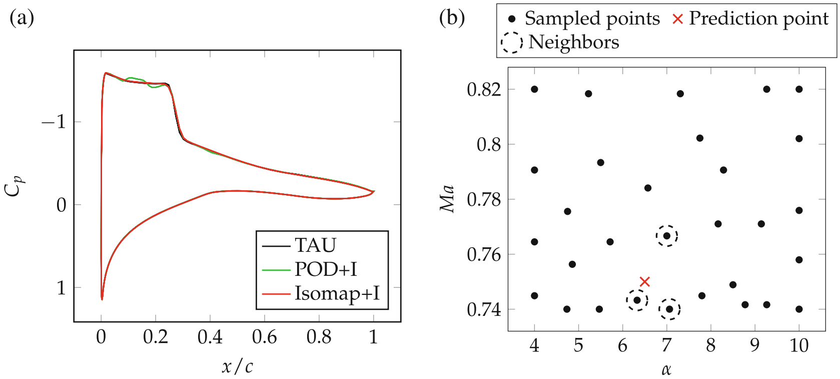


Fig. 16.9 Prediction of the surface C_p-distribution at $(\alpha, \mathrm{Ma}) = (6.5°, 0.75)$ based on 5 initial plus 25 adaptively sampled snapshots via HYE. (**a**) Prediction of the surface C_p-distribution at $(\alpha, \mathrm{Ma}) = (6.5°, 0.75)$. (**b**) HYE based final sampling

only three neighboring snapshots (see Fig. 16.9b). The POD+I based prediction only shows a small mismatch upstream of the shock.

16.5 Conclusions

We have developed two adaptive sampling strategies, referred to as the maximum distance error (MDE) and the hybrid error (HYE) strategy, respectively, that aim at determining sample locations in a given parameter domain of interest such that a well-distributed homogeneous design of experiment is achieved in the embedding space with as few high-fidelity sample computations as possible. The underlying assumption is that the sample data is contained in a submanifold of low intrinsic dimension that is embedded in a large-dimensional Euclidean vector space. Thus, the notions of 'well-distributed' and 'homogeneous' are to be understood with respect to the geometry of this submanifold.

Both adaptive sampling methods try to generate manifold filling sample data sets such that the essential geometric characteristics of the underlying submanifold are captured. The MDE strategy relies on the geodesic interpoint distances that are approximated using the Isomap manifold learning technique. The HYE strategy additionally considers the reconstruction error of an Isomap+I emulator during the sampling process, such that the sample density in the highly nonlinear regions of the manifold, where the error is expected to be larger, is augmented.

In the numerical experiments, we have shown that the adaptive sampling strategies eventually lead to more accurate emulators than when using space filling random samplings of the same cardinality. More precisely, the advantages over random samplings have been demonstrated for an Isomap-based emulator of the viscous flow around the 2D NACA 64A010 airfoil. Moreover, we observed that the standard POD-based flow emulator also benefits from the Isomap-induced adaptive sampling process.

References

1. Allmaras, S.R., Johnson, F.T.: Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. In: Seventh International Conference on Computational Fluid Dynamics (ICCFD7), pp. 1–11 (2012)
2. Bernstein, A.V., Kuleshov, A.P.: Tangent bundle manifold learning via Grassmann & Stiefel eigenmaps. CoRR, abs/1212.6031 (2012)
3. Blazek, J.: Computational Fluid Dynamics: Principles and Applications, 1st edn. Elsevier, Amsterdam/London/New York/Oxford/Paris/Shannon/Tokyo (2001)
4. Bui-Thanh, T., Damodaran, M., Willcox, K.: Proper orthogonal decomposition extensions for parametric applications in transonic aerodynamics. AIAA J. **42**(8), 1505–1516 (2004)
5. Cayton, L.: Algorithms for manifold learning. University of California at San Diego Tech. Rep, pp. 1–17 (2005)

6. Forrester, A., Sóbester, A., Keane, A.: Engineering Design via Surrogate Modelling: A Practical Guide. Wiley, London (2008)
7. Franz, T.: Reduced-order modeling for steady transonic flows via manifold learning. PhD thesis, TU Braunschweig (2016)
8. Franz, T., Zimmermann, R., Görtz, S., Karcher, N.: Interpolation-based reduced-order modeling for steady transonic flows via manifold learning. Int. J. Comput. Fluid Mech. (Special Issue on Reduced Order Modeling) **228**, 106–121 (2014)
9. Halton, J.H.: On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. Numer. Math. **2**(1), 84–90 (1960)
10. Langer, S., Schwöppe, A., Kroll, N.: The DLR flow solver TAU—status and recent algorithmic developments. In: 52nd AIAA Aerospace Sciences Meeting. AIAA Paper, vol. 80 (2014)
11. Mardia, K.V., Kent, J.T., Bibby, J.M.: Multivariate Analysis. Academic (1979)
12. Pinnau, R.: Model reduction via proper orthogonal decomposition. In: Schilders, W.H.A., van der Vorst, H.A., Rommes, J. (eds.) Model Order Reduction: Theory, Research Aspects and Applications. Mathematics in Industry, vol. 13, pp. 95–109. Springer, Berlin/Heidelberg (2008)
13. Powell, M.J.D.: Radial basis function methods for interpolation to functions of many variables. In: Proceedings of Hellenic European Conference on Computer Mathematics and Its Applications (HERCMA), pp. 2–24 (2001)
14. Rajan, V.T.: Optimality of the Delaunay triangulation in $\mathbb{R}^d$. Discrete Comput. Geom. **12**(1), 189–202 (1994)
15. Rozza, G., Huynh, D.B.P., Patera, A.T.: Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics. Arch. Comput. Methods Eng. **15**, 229–275 (2008)
16. Spivak, M.: A Comprehensive Introduction to Differential Geometry, vol. 1, 3rd edn. Publish or Perish (1999)
17. Tenenbaum, J.B., De Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**(5500), 2319–2323 (2000)
18. Van der Maaten, L.J.P., Postma, E.O., Van den Herik, H.J.: Dimensionality reduction: a comparative review. J. Mach. Learn. Res. **10**, 1–41 (2009)
19. Zimmermann, R., Görtz, S.: Improved extrapolation of steady turbulent aerodynamics using a non-linear POD-based reduced order model. Aeronaut. J. **116**(1184), 1079–1100 (2012)

Chapter 17
Cross-Gramian-Based Model Reduction: A Comparison

Christian Himpe and Mario Ohlberger

Abstract As an alternative to the popular balanced truncation method, the cross Gramian matrix induces a class of balancing model reduction techniques. Besides the classical computation of the cross Gramian by a Sylvester matrix equation, an empirical cross Gramian can be computed based on simulated trajectories. This work assesses the cross Gramian and its empirical Gramian variant for state-space reduction on a procedural benchmark based on the cross Gramian itself.

17.1 Introduction

The cross Gramian matrix is an interesting mathematical object with manifold applications in control theory, system theory and even information theory [11]. Yet, first and foremost the cross Gramian is used in the context of model order reduction.

The cross Gramian was introduced in [5] for SISO (Single-Input-Single-Output) systems and extended in [6, 18] to MIMO (Multiple-Input-Multiple-Output) systems as an alternative balancing method to the balanced truncation [22] model reduction technique. A data-driven variant of the cross Gramian, the empirical cross Gramian, was proposed in [29] for SISO systems and extended in [10] to MIMO systems, expanding the set of empirical Gramians [15, 16].

Various approaches for cross-Gramian-based model reduction have been studied [1, 10, 24, 27, 28]. This work compares a small selection of these methods, using a procedural benchmark based on a method to generate random systems introduced in [26]. In this setting, a linear time-invariant input-output system is the central object of interest:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{17.1}$$

C. Himpe (✉) • M. Ohlberger
Institute for Computational and Applied Mathematics, University of Münster, Einsteinstrasse 62, 48149 Münster, Germany
e-mail: christian.himpe@uni-muenster.de; mario.ohlberger@uni-muenster.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_17

which consists of a dynamical system and an output equation. The associated vector field is given as a linear transformation of the state $x : \mathbb{R} \to \mathbb{R}^N$ by the system matrix $A \in \mathbb{R}^{N \times N}$, and a source term introducing the input $u : \mathbb{R} \to \mathbb{R}^M$ through the input matrix $B \in \mathbb{R}^{N \times M}$. In the scope of this work the real parts of the eigenvalues of A are assumed to be negative which implies an asymptotically stable dynamical system. The output $y : \mathbb{R} \to \mathbb{R}^Q$ is determined by an output functional consisting of a linear transformation of the state x by the output matrix $C \in \mathbb{R}^{Q \times N}$, and a term forwarding the input u by the feed-through matrix $D \in \mathbb{R}^{Q \times M}$; the latter is assumed to be trivial $D = 0$ in this contribution, as it does not affect the investigated model reduction procedures.

This work is structured as follows: An outline of the cross Gramian, the empirical cross Gramian and the considered methods for cross-Gramian-based model reduction is given in Sect. 17.2. In Sect. 17.3 the procedural benchmark is proposed, and in Sect. 17.4 the considered methods are tested upon this benchmark.

17.2 The Cross Gramian

Two operators play a central role in systems theory [14]: The controllability operator $\mathscr{C} : L_2^M \to \mathbb{R}^N$ and the observability operator $\mathscr{O} : \mathbb{R}^N \to L_2^Q$:

$$\mathscr{C}(u) = \int_{-\infty}^{0} \mathrm{e}^{-At} Bu(t) \mathrm{d}t, \qquad \mathscr{O}(x_0) = C \, \mathrm{e}^{At} x_0;$$

the former measures how much energy introduced by u is needed to drive x to a certain state, the latter quantifies how well the state x is visible in the output y. A composition of the observability with the controllability operator yields the Hankel operator $H : L_2^M \to L_2^Q$,

$$H = \mathscr{O} \circ \mathscr{C},$$

whose singular values, the so called Hankel singular values, classify the states by importance in terms of the system's input-output coherence. Commonly, the action of the Hankel operator is described by "mapping past inputs to future outputs" [8].

The permuted composition of $\mathscr{C}$ with $\mathscr{O}$, that is only admissible for square systems, which have the same number of inputs and outputs $M = Q$, yields a cross operator $W_X : \mathbb{R}^N \to \mathbb{R}^N$.

Definition 1 The composition of the controllability operator $\mathscr{C}$ with the observability operator $\mathscr{O}$ is called **cross Gramian** W_X:

$$W_X := \mathscr{C} \circ \mathscr{O} = \int_0^{\infty} \mathrm{e}^{At} BC \, \mathrm{e}^{At} \, \mathrm{d}t \in \mathbb{R}^{N \times N}.$$

This cross Gramian concurrently encodes controllability and observability information of the underlying system. Despite the name, the cross Gramian is generally neither symmetric nor positive semi-definite, hence, it is not a Gramian matrix but it was introduced under this name in [5].

An obvious connection between the Hankel operator and the cross Gramian is given by the equality of their traces:

$$\mathrm{tr}(H) = \mathrm{tr}(\mathscr{O}\mathscr{C}) = \mathrm{tr}(\mathscr{C}\mathscr{O}) = \mathrm{tr}(W_X).$$

Similarly, the logarithm-determinants are equal: $\mathrm{logdet}(H) = \mathrm{logdet}(W_X)$, which is the basis for the cross-Gramian-based information index [7] measuring information entropy. Yet, a central property of the cross Gramian is only available for symmetric systems.

Lemma 1 *For a symmetric system the absolute values of the eigenvalues of the cross Gramian are equal to the Hankel singular values:*

$$\sigma_i(H) = |\lambda_i(W_X)|.$$

This property is expanded to orthogonally symmetric systems in [4].

Proof A symmetric system has a symmetric Hankel operator:

$$H = H^* \Rightarrow \mathscr{O}\mathscr{C} = (\mathscr{O}\mathscr{C})^*.$$

Hence, for the singular values of the Hankel operator holds:

$$\begin{aligned}\sigma_i(H) = \sigma_i(\mathscr{O}\mathscr{C}) &= \sqrt{\lambda_i(\mathscr{O}\mathscr{C}(\mathscr{O}\mathscr{C})^*)} = \sqrt{\lambda_i(\mathscr{O}\mathscr{C}\mathscr{O}\mathscr{C})} \\ &\overset{[13]}{=} \sqrt{\lambda_i(\mathscr{C}\mathscr{O}\mathscr{C}\mathscr{O})} = \sqrt{\lambda_i(W_X W_X)} = |\lambda_i(W_X)|.\end{aligned}$$

□

Classically, to compute the cross Gramian, a relation to the solution of a matrix equation is exploited.

Lemma 2 *The cross Gramian is the solution to the Sylvester matrix equation:*

$$AW_X + W_X A = -BC. \tag{17.2}$$

Proof This is a special case of [17, Theorem 5]

17.2.1 The Empirical Cross Gramian

An alternative approach to the computation of the cross Gramian via a matrix equation is the computation of its empirical variant. Empirical (controllability and observability) Gramians were first introduced in [15, 16] and result from (numerically obtained) trajectory data. Following, a summary of the empirical cross Gramian [10], which extends ideas from [15, 29] is given. A justification for using an empirical Gramian is given i.e. by the definition of the cross Gramian,

$$W_X = \int_0^\infty (\mathrm{e}^{At} B)(\mathrm{e}^{A^\intercal t} C^\intercal)^\intercal \mathrm{d}t,$$

which can be interpreted as cross covariance matrix of the system's impulse response and adjoint system's impulse response. As originally in [22], these impulse responses are trajectories,

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B\delta(t) \Rightarrow x(t) = \mathrm{e}^{At} B, \\ \dot{z}(t) &= A^\intercal z(t) + C^\intercal \delta(t) \Rightarrow z(t) = \mathrm{e}^{A^\intercal t} C^\intercal, \\ \Rightarrow W_X &= \int_0^\infty x(t) z(t)^\intercal \mathrm{d}t, \end{aligned} \tag{17.3}$$

and yield an **empirical linear cross Gramian** [2, Sect. 2.3].

A more general definition of the empirical cross Gramian [10, 29], without relying on the linear structure of the underlying system, such as a closed form for the adjoint system, is then even applicable to nonlinear systems.

Definition 2 For sets $\{c_k \in \mathbb{R} \setminus 0 : l = 1 \dots K\}$, $\{d_k \in \mathbb{R} \setminus 0 : l = 1 \dots L\}$, the m-th M-dimensional standard base vector $e_{M,m}$ and the j-th N-dimensional standard base vector $e_{N,j}$, the **empirical cross Gramian** $\widehat{W}_X \in \mathbb{R}^{N \times N}$ is given by:

$$\begin{aligned} \widehat{W}_X &:= \frac{1}{KLM} \sum_{k=1}^{K} \sum_{l=1}^{L} \sum_{m=1}^{M} \frac{1}{c_k d_l} \int_0^\infty \Psi^{klm}(t) \mathrm{d}t, \\ \Psi_{ij}^{klm}(t) &= \left(x_i^{km}(t) - \bar{x}_i^{km}\right) \left(y_m^{lj}(t) - \bar{y}_m^{lj}\right), \end{aligned} \tag{17.4}$$

with x_i^{km} being the i-th component of the state trajectory for the input $u^{km}(t) = c_k e_{M,m} \delta(t)$, given initial state and $\bar{x}_i^{km}$ the associated temporal average state, while y_m^{lj} is the m-th component of the output trajectory for the initial state $x_0^{lj} = d_l e_{N,j}$, zero input and $\bar{y}^{lj}$ the associated temporal average output.
This empirical cross Gramian requires $K \cdot M$ state trajectories for perturbed impulse input, and $L \cdot N$ output trajectories for perturbed initial states with no input. The sets $\{c_k\}$ and $\{d_l\}$ define the operating region of the underlying system and determine for which inputs and initial states the empirical cross Gramian is valid. In [10] the empirical cross Gramian is generalized to an empirical cross covariance matrix by admitting arbitrary input functions and centering the state and output trajectories about their respective steady state. Furthermore, it is shown in [10] that the empirical cross Gramian is equal to the cross Gramian in Definition 1 for linear systems (17.1).

Theorem 1 *For an asymptotically stable linear system, the empirical cross Gramian $\widehat{W}_X$ reduces to the cross Gramian W_X.*

Proof See [10, Lemma 3]

Since this empirical cross Gramian requires merely discrete (output) trajectories and does not rely on the linear $\Sigma(A, B, C)$ structure of the system, it can be computed also for nonlinear systems. Due to the only prerequisite of trajectory data, empirical Gramians are a flexible tool, but warrant prior knowledge on the operating region of the system to define the perturbations. Hence, based on the idea of numerical linearization cf. [21], empirical Gramians give rise to a data-driven nonlinear model reduction technique.

The empirical cross Gramian consists of inner products between state trajectories with perturbed input and output trajectories with perturbed initial state. This allows, by treating the parameters as additional (constant) states, to extend the cross Gramian beyond state input-output coherence to include observability-based parameter identifiability information [10]. The associated empirical joint Gramian is an empirical cross Gramian that enables a combined state and parameter reduction from a single cross operator.

Furthermore, a cross Gramian for non-symmetric and also non-square systems [12], which can be efficiently computed in its empirical variant, expands the applicability of the cross Gramian to more general system configurations.

17.2.2 Cross-Gramian-Based Model Reduction

Model Reduction is the principal application of the cross Gramian. Cross-Gramian-based model reduction is a projection-based approach: The state-space trajectory is approximated by a lower-dimensional trajectory, which results from a reducing truncated projection $R \in \mathbb{R}^{n \times N}$ and a reconstructing truncated projection $S \in \mathbb{R}^{N \times n}$ for $n < N$:

$$x_r(t) := Rx(t) \Rightarrow x(t) \approx Sx_r(t).$$

Using such projections, a reduced order model for the full order system is given by:

$$\begin{aligned}\dot{x}_r(t) &= RASx_r(t) + RBu_r(t),\\ y_r(t) &= CSx_r(t),\end{aligned}$$

and $x_{r,0} = Rx_0$. This can be simplified by $A_r := RAS$, $B_r := RB$, $C_r := CS$, due to the linear structure of the system:

$$\begin{aligned}\dot{x}_r(t) &= A_r x_r(t) + B_r u(t),\\ y_r(t) &= C_r x_r(t).\end{aligned}$$

To obtain such projections from the cross Gramian, various methods can be used. The eigenvalue decomposition of the cross Gramian matrix,

$$W_X \overset{\text{EVD}}{=} T\Lambda T^{-1},$$

given a symmetric system, yields a balancing projection $S := T$, $R := T^{-1}$ [1], which can be truncated based on the absolute value of the magnitude of the eigenvalues $|\lambda_i| = |\Lambda_{ii}|$. Alternatively, a singular value decomposition of the cross Gramian,

$$W_X \overset{\text{SVD}}{=} U\Sigma V,$$

can be utilized. Similarly, $S := U$ and $R := V$ can be truncated based on the associated singular values $\sigma_i = \Sigma_{ii}$; yet this projection is only approximately balancing [24, 28] and the reduced order model's stability is not guaranteed to be preserved.

As a variant, only the left or right singular vectors can be used individually as a Galerkin projection,

$$\begin{cases} S := U & R := U^\intercal \\ S := V^\intercal & R := V. \end{cases} \tag{17.5}$$

This direct truncation [10] is less accurate, but provides an orthogonal projection.

Lastly, we note that instead of truncating the decomposition derived projections based on the eigen- or singular values, it is suggested in [3], to use the quantities $d_i := |\tilde{b}_i\tilde{c}_i\lambda_i|$ and $\hat{d}_i := |\tilde{b}_i\tilde{c}_i\sigma_i|$ (compare (17.6)) for balanced and approximately balanced systems respectively, which utilizes the columns of the (approximately) balanced input matrix $\tilde{b}_i$ and rows of the (approximately) balanced output matrix $\tilde{c}_i$.

17.3 Inverse Sylvester Procedure

To compare the cross-Gramian-based reduced order model quality of the classically computed cross Gramian (17.2) and the empirical cross Gramian from Sect. 17.2.1, a procedural and randomly generated benchmark system of variable state-space dimension is presented. The proposed system generator is a special case of the inverse Lyapunov procedure [26]. This variant though generates exclusively state-space symmetric systems featuring $A = A^\intercal$ and $B = C^\intercal$, which are found in applications such as RC circuits and have some interesting properties as shown [19, 23], such as equality of the controllability, observability and cross Gramian: $W_C = W_O = W_X$.

We note that the cross Gramian, as an $N \times N$ dimensional linear operator $W_X : \mathbb{R}^N \to \mathbb{R}^N$, is an endomorphism. This leads to the following relation between the system matrix A and the cross Gramian matrix W_X, as stated in [20]:

Corollary 1 *Let W_X be the cross Gramian to the system $\Sigma(A, B, C)$. Then A is **a** cross Gramian to the virtual system $(-W_X, B, C)$.*

Proof This is a direct consequence of Lemma 2.
Hence, for a known cross Gramian W_X, input matrix B and output matrix C, an associated system matrix A can be computed as the cross Gramian of the virtual system. To ensure the (asymptotic) stability of the system, an observation from [19, Theorem 2.1] is utilized.

Lemma 3 *For a state-space symmetric system the cross Gramian is symmetric and positive semi-definite.*

Proof Given a state-space symmetric system, the associated cross Gramian's Sylvester equation (17.2) becomes a Lyapunov equation:

$$AW_X + W_X A = BC \Leftrightarrow AW_X + W_X A^\intercal = BB^\intercal,$$

of which a solution is symmetric and positive semi-definite. □
Thus, an (asymptotically) stable state-space symmetric system can be generated by providing an input matrix B, which determines the output matrix $C = B^\intercal$ and a symmetric positive semi-definite cross Gramian W_X. A procedure[1] to generate random asymptotically stable state-space symmetric systems, called **inverse Sylvester procedure**, is given by:

1. Sample the cross Gramian's eigenvalues to define a positive definite cross Gramian in balanced form, which is a diagonal matrix, from $\lambda_i = a(\frac{b}{a})^{\mathscr{U}_{[0,1]}}$ with $0 < a < b$.
2. Sample an input matrix B from an iid multivariate standard normal distribution $\mathscr{N}_{0,1}^{N \times M}$ and set the output matrix to $C := B^\intercal$.
3. Solve $-W_X A - AW_X = -BC \Leftrightarrow W_X A + AW_X = BC$ for (a negative semi-definite) system matrix A.
4. Sample an orthogonal (un-)balancing transformation U by a QR decomposition of a multivariate standard normally distributed matrix $\{U, R\} = \mathrm{qr}(\mathscr{N}_{0,1}^{N \times N})$.
5. Unbalance the system by: $U^\intercal AU$, $U^\intercal B$, CU

17.4 Model Reduction Experiments

In this section the Sylvester-equation-based cross Gramian is compared to the empirical cross Gramians in terms of state-space model reduction of a random system generated by the inverse Sylvester procedure. A test system is generated

[1] See also `isp.m` in the associated source code archive.

as state-space symmetric SISO system, $M = Q = 1$, of order $N = 1000$ by the inverse Sylvester procedure, using $a = \frac{1}{10}$, $b = 10$, excited by zero-mean, unit-variance Gaussian noise during each time-step and starting from a zero initial state. Due to the use of empirical Gramians, a time horizon of $T = 1$ and a fixed time-step width of $h = \frac{1}{100}$ is selected. The cross Gramian variants are computed by solving a matrix equation (17.2), by the empirical linear cross Gramian (17.3) [9] and by the empirical cross Gramian (17.4) [9], from which the reducing projections are obtained using the direct truncation approximate balancing (17.5) method. During the construction of the empirical cross Gramians an impulse input $u(t) = \delta(t)$ is utilized. The state-space symmetry implies, first, that the matrix equation for the cross Gramian is practically a Lyapunov equation, and second, that utilizing the SVD of the cross Gramian is equivalent to the eigendecomposition.

The model reduction error, the error between the FOM (Full Order Model) and ROM (Reduced Order Model) output, is measured in the (time-domain) Lebesgue L_1-, L_2- and L_∞-norms,

$$\begin{aligned} \|y - y_r\|_{L_1} &= \int_0^\infty \|y(t) - y_r(t)\|_1 \mathrm{d}t, \\ \|y - y_r\|_{L_2} &= \sqrt{\int_0^\infty \|y(t) - y_r(t)\|_2^2 \mathrm{d}t}, \\ \|y - y_r\|_{L_\infty} &= \operatorname*{ess\,sup}_{t \in [0,\infty)} \|y(t) - y_r(t)\|_\infty, \end{aligned}$$

as well as approximately in the (frequency-domain) Hardy H_∞-norm and approximately in the Hardy H_2-norm. Since a state-space symmetric SISO system is used, twice the truncated tail of singular values not only bounds, but equals the H_∞-error between the original and reduced transfer function G and G_r [19, Theorem 4.1]:

$$\|G - G_r\|_{H_\infty} = 2 \sum_{i=n+1}^{N} \sigma_i.$$

Thus, the H_∞-error of the reduced order model can be approximated by this formula using the singular values of a numerically approximated cross Gramian.

The H_2-error is approximated based on [28, Remark 3.3]:

$$\|G - G_r\|_{H_2} \approx \sqrt{\operatorname{tr}(\widetilde{C}_2 W_{X,22} \widetilde{B}_2)}, \tag{17.6}$$

with the balanced and truncated input and output matrices $\widetilde{B}_2 = B - UU^\intercal B$ and $\widetilde{C}_2 = C - CUU^\intercal$ as well as the truncated square lower right block of the balanced diagonal cross Gramian $W_{X,22}$.

Figures 17.1, 17.2 and 17.3 show the relative L_1-, L_2- and L_∞-output errors for the classic cross Gramian ($W_{X,1}$), empirical linear cross Gramian ($W_{X,2}$) and

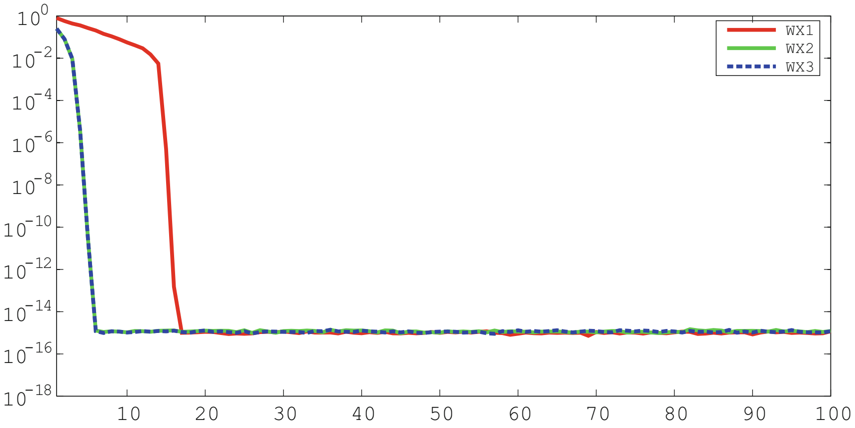


Fig. 17.1 Relative L_1 output error between the FOM and ROMs for the matrix equation based cross Gramian $W_{X,1}$, the empirical linear cross Gramian $W_{X,2}$ and the empirical cross Gramian $W_{X,3}$

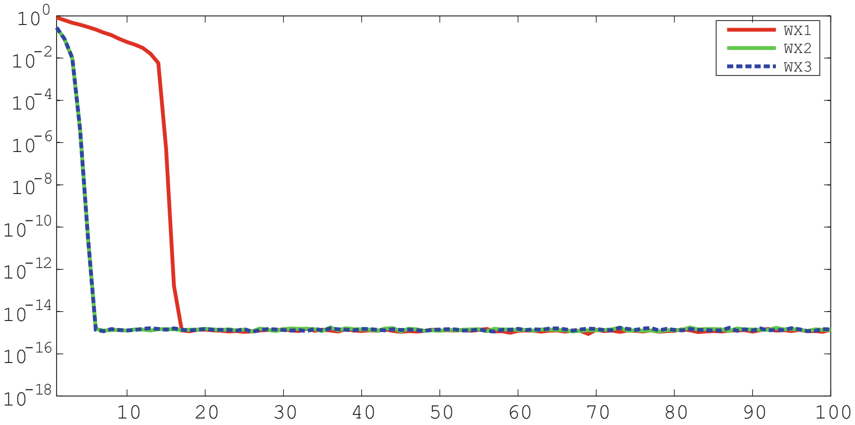


Fig. 17.2 Relative L_2 output error between the FOM and ROMs for the matrix equation based cross Gramian $W_{X,1}$, the empirical linear cross Gramian $W_{X,2}$ and the empirical cross Gramian $W_{X,3}$

empirical cross Gramian ($W_{X,3}$) using approximate-balancing-based projections over varying reduced state-space dimension up to order $\dim(x_r(t)) = 100$. These errors are approximately computed, based on the previous definitions, by simulating the full and reduced order model with fixed input time series of Gaussian noise.

For all tested cross Gramians, the Lebesgue error measures behave very similarly. While the output errors for the empirical linear cross Gramian and the empirical cross Gramian decay at a reduced order of $n \geq 7$ to a level near the numerical precision with a similar rate, the matrix equation derived cross Gramian reaches this

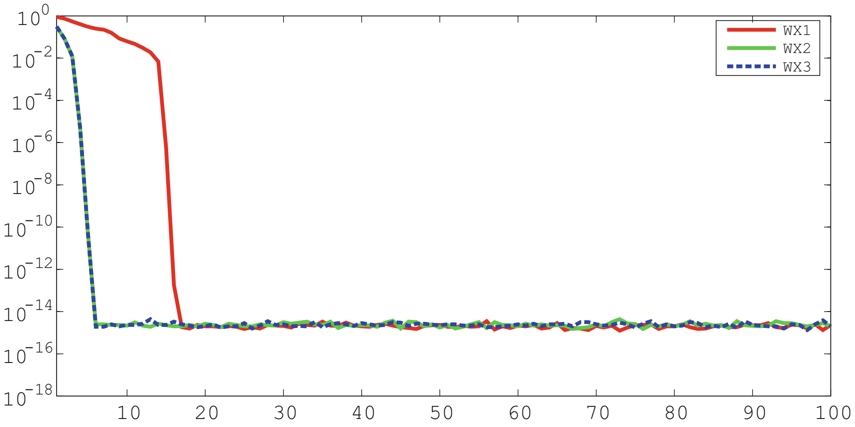


Fig. 17.3 Relative L_∞ output error between the FOM and ROMs for the matrix equation based cross Gramian $W_{X,1}$, the empirical linear cross Gramian $W_{X,2}$ and the empirical cross Gramian $W_{X,3}$

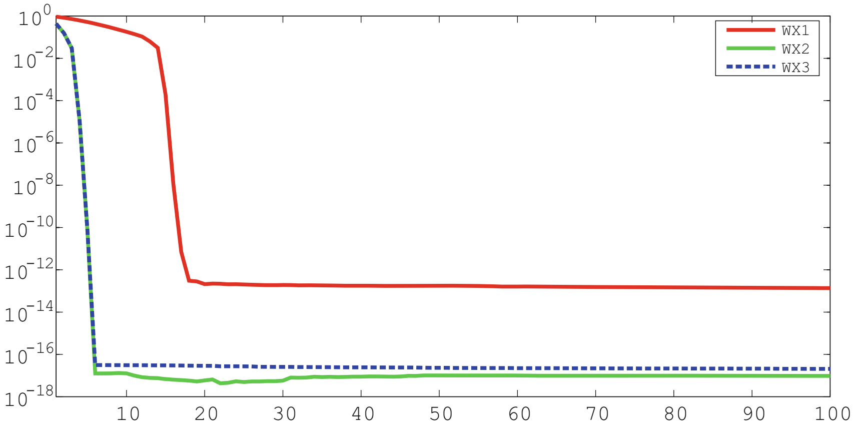


Fig. 17.4 Approximate relative H_2 output error between the FOM and ROMs for the matrix equation based cross Gramian $W_{X,1}$, the empirical linear cross Gramian $W_{X,2}$ and the empirical cross Gramian $W_{X,3}$

level at $n \geq 18$. Overall, the model reduces very well and due to the specific time frame for the reduction and comparison and the empirical Gramians yield better results.

In Figs. 17.4 and 17.5 the approximate H_2-error and the approximate H_∞-error are depicted for the three cross Gramian variants over varying reduced orders up to $\dim(x_r(t)) = 100$.

For the frequency-domain errors the cross Gramian obtained as solution to a Sylvester (Lyapunov) equation does not attain the same accuracy as the empirical

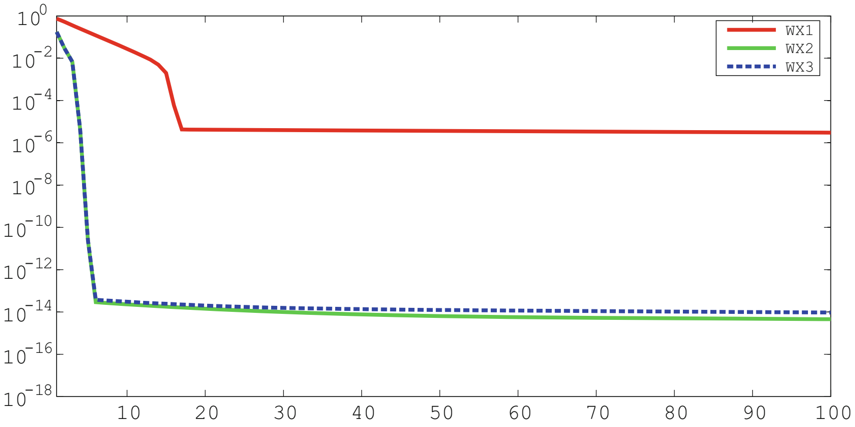


Fig. 17.5 Approximate relative H_∞ output error between the FOM and ROMs for the matrix equation based cross Gramian $W_{X,1}$, the empirical linear cross Gramian $W_{X,2}$ and the empirical cross Gramian $W_{X,3}$

cross Gramians, which reach the numerical precision level for $n \geq 6$. Also, as for the time-domain errors, the sharp decay in the output error occurs at a higher reduced order $n \geq 19$ for the non-empirical cross Gramian, but machine precision is not reached.

The difference in reduced order model quality in the time-domain norms between the empirically and non-empirically computed cross Gramians arises from the restricted operating region, especially in terms of the numerical integration order, time-step width and time horizon. For the frequency-domain norms, the disparity in accuracy originates in the different computational approaches of numerically approximating the cross Gramian matrix, as the Hardy-norm computations utilize the respective cross Gramian's singular values.

17.5 Conclusion

This work summarized the cross Gramian and its empirical variant and assessed methods for cross-Gramian-based model reduction mathematically and numerically. The latter is conducted by a new cross-Gramian-based random state-space symmetric system generator. Due to the strict definition of the operating region of the test system, the empirical cross Gramians produce superior reduced order models. This confirms the results of [25], that empirical Gramians can convey more information on the input-output behavior for a specific operating region than the classic matrix equation approach.

17.6 Code Availability

The source code of the implementations used to compute the presented results can be obtained from:

http://www.runmycode.org/companion/view/1854
and is authored by: Christian Himpe.

Acknowledgements This work was supported by the Deutsche Forschungsgemeinschaft: DFG EXC 1003 Cells in Motion - Cluster of Excellence, Münster, Germany and by the Center for Developing Mathematics in Interaction, DEMAIN, Münster, Germany.

References

1. Aldhaheri, R.W.: Model order reduction via real Schur-form decomposition. Int. J. Control **53**(3), 709–716 (1991)
2. Baur, U., Benner, P., Haasdonk, B., Himpe, C., Martini, I., Ohlberger, M.: Comparison of methods for parametric model order reduction of instationary problems. In: Benner, P., Cohen, A., Ohlberger, M., Willcox, K. (eds.) Model Reduction and Approximation: Theory and Algorithms. SIAM, Philadelphia (2017)
3. Davidson, A.: Balanced system and model reduction. Electron. Lett. **22**(10), 531–532 (1986)
4. De Abreu-Garcia, J.A., Fairman, F.W.: A note on cross Grammians for orthogonally symmetric realizations. IEEE Trans. Autom. Control **31**(9), 866–868 (1986)
5. Fernando, K.V., Nicholson, H.: On the structure of balanced and other principal representations of SISO systems. IEEE Trans. Autom. Control **28**(2), 228–231 (1983)
6. Fernando, K.V., Nicholson, H.: On the cross-Gramian for symmetric MIMO systems. IEEE Trans. Circuits Syst. **32**(5), 487–489 (1985)
7. Fu, J.-B., Zhang, H., Sun, Y.-X.: Model reduction by minimizing information loss based on cross-Gramian matrix. J. Zhejiang Univ. (Eng. Sci.) **43**(5), 817–821 (2009)
8. Glover, K.: All optimal Hankel-norm approximations of linear multivariable systems and their L^{∞}-error bounds. Int. J. Control **39**(6), 1115–1193 (1984)
9. Himpe, C.: emgr - Empirical Gramian Framework (Version 3.9). http://gramian.de (2016)
10. Himpe, C., Ohlberger, M.: Cross-Gramian based combined state and parameter reduction for large-scale control systems. Math. Probl. Eng. **2014**, 1–13 (2014)
11. Himpe, C., Ohlberger, M.: The versatile cross Gramian. In: ScienceOpen Posters, vol. MoRePas 3 (2015)
12. Himpe, C., Ohlberger, M.: A note on the cross Gramian for non-symmetric systems. Syst. Sci. Control Eng. **4**(1), 199–208 (2016)
13. Hladnik, M., Omladič, M.: Spectrum of the product of operators. Proc. Am. Math. Soc. **102**(2), 300–302 (1988)
14. Kalman, R.E.: Mathematical description of linear dynamical systems. J. Soc. Ind. Appl. Math. A Control **1**(2), 152–192 (1963)
15. Lall, S., Marsden, J.E., Glavaski, S.: Empirical model reduction of controlled nonlinear systems. In: Proceedings of the 14th IFAC Congress, vol. F, pp. 473–478 (1999)
16. Lall, S., Marsden, J.E., Glavaski, S.: A subspace approach to balanced truncation for model reduction of nonlinear control systems. Int. J. Robust Nonlinear Control **12**(6), 519–535 (2002)
17. Lancaster, P.: Explicit solutions of linear matrix equations. SIAM Rev. **12**(4), 544–566 (1970)
18. Laub, A.J., Silverman, L.M., Verma, M.: A note on cross-Grammians for symmetric realizations. Proc. IEEE **71**(7), 904–905 (1983)

19. Liu, W.Q., Sreeram, V., Teo, K.L.: Model reduction for state-space symmetric systems. Syst. Control Lett. **34**(4), 209–215 (1998)
20. Mironovskii, L.A., Solov'eva, T.N.: Analysis of multiplicity of Hankel singular values of control systems. Autom. Remote Control **76**(2), 205–218 (2015)
21. Moore, B.C.: Principal component analysis in nonlinear systems: preliminary results. In: 18th IEEE Conference on Decision and Control including the Symposium on Adaptive Processes, vol. 2, pp. 1057–1060 (1979)
22. Moore, B.: Principal component analysis in linear systems: controllability, observability, and model reduction. IEEE Trans. Autom. Control **26**(1), 17–32 (1981)
23. Opmeer, M.R., Reis, T.: The balanced truncation error bound in Schatten norms. Hamburger Beiträge zur Angewandten Math. **2014**(1), 1–6 (2014)
24. Rahrovani, S., Vakilzadeh, M.K., Abrahamsson, T.: On Gramian-based techniques for minimal realization of large-scale mechanical systems. In: Topics in Modal Analysis, vol. 7, pp. 797–805. Springer, New York (2014)
25. Singh, A.K., Hahn, J.: On the use of empirical Gramians for controllability and observability analysis. In: Proceedings of the American Control Conference, vol. 2005, pp. 140–141 (2005)
26. Smith, S.C., Fisher, J.: On generating random systems: a Gramian approach. In: Proceedings of the American Control Conference, vol. 3, pp. 2743–2748 (2003)
27. Sorensen, D.C., Antoulas, A.C.: Projection methods for balanced model reduction. Technical report, Rice University, 2001
28. Sorensen, D.C., Antoulas, A.C.: The Sylvester equation and approximate balanced reduction. Linear Algebra Appl. **351–352**, 671–700 (2002)
29. Streif, S., Findeisen, R., Bullinger, E.: Relating cross Gramians and sensitivity analysis in systems biology. In: Proceedings of the Theory of Networks and Systems, vol. 10.4, pp. 437–442 (2006)

Chapter 18
Truncated Gramians for Bilinear Systems and Their Advantages in Model Order Reduction

Peter Benner, Pawan Goyal, and Martin Redmann

Abstract In this paper, we discuss truncated Gramians (TGrams) for bilinear control systems and their relations to Lyapunov equations. We show how TGrams relate to input and output energy functionals, and we also present interpretations of controllability and observability of the bilinear systems in terms of these TGrams. These studies allow us to determine those states that are less important for the system dynamics via an appropriate transformation based on the TGrams. Furthermore, we discuss advantages of the TGrams over the Gramians for bilinear systems as proposed in Al-baiyat and Bettayeb (Proceedings of 32nd IEEE CDC, pp. 22–27, 1993). We illustrate the efficiency of the TGrams in the framework of model order reduction via a couple of examples, and compare to the approach based on the full Gramians for bilinear systems.

18.1 Introduction

Direct numerical simulations are one of the conventional methods to study physical phenomena of dynamical systems. However, extracting all the complex system dynamics generally leads to large state-space dynamical systems, whose direct simulations are inefficient and involve a huge computational burden. Hence, there is a need to consider *model order reduction* (MOR), aiming to replace these large-scale dynamical systems by systems of much smaller state dimension. MOR for linear systems has been investigated intensively in recent years and is widely used in numerous applications; see, e.g., [2, 8, 23]. In this work, we consider MOR for

P. Benner
Computational Methods in Systems and Control Theory, Max Planck Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany
e-mail: benner@mpi-magdeburg.mpg.de

P. Goyal (✉) • M. Redmann
Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstr. 1, 39106 Magdeburg, Germany
e-mail: goyalp@mpi-magdeburg.mpg.de; redmann@mpi-magdeburg.mpg.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_18

bilinear control systems, which can be considered as a bridge between linear and nonlinear systems and are of the form:

$$\begin{aligned} \dot{x}(t) &= Ax(t) + \sum\nolimits_{k=1}^{m} N^{(k)}x(t)u_k(t) + Bu(t), \\ y(t) &= Cx(t), \quad x(0) = 0, \end{aligned} \tag{18.1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ and $y(t) \in \mathbb{R}^p$ are the state, input and output vectors of the system, respectively. The numbers m and p represent the quantity of inputs and outputs. All system matrices are of appropriate dimensions. The applications of bilinear systems can be seen in various fields [11, 18, 21]. Further, the applicability of the systems (18.1) in MOR for stochastic control problems is studied in [7, 17] and for MOR of a certain class of linear parametric systems in [4]. Our goal is to construct another low-dimensional bilinear system

$$\begin{aligned} \dot{\hat{x}}(t) &= \hat{A}\hat{x}(t) + \sum\nolimits_{k=1}^{m} \hat{N}^{(k)}\hat{x}(t)u_k(t) + \hat{B}u(t), \\ \hat{y}(t) &= \hat{C}\hat{x}(t), \quad \hat{x}(0) = 0, \end{aligned} \tag{18.2}$$

where $\hat{A}, \hat{N}^{(k)} \in \mathbb{R}^{r\times r}$, $\hat{B} \in \mathbb{R}^{r\times m}$ and $\hat{C} \in \mathbb{R}^{p\times r}$ with $r \ll n$, ensuring $y \approx \hat{y}$ for all admissible inputs $u \in L^2(0,\infty)$ with components u_k, $k = 1,\dots,m$. Analogous to linear systems, we aim to obtain the reduced matrices via projection. For this, we construct two projection matrices $V, W \in \mathbb{R}^{n\times r}$ such that W^TV is invertible, which allow us to determine the reduced matrices as:

$$\begin{aligned} \hat{A} &= (W^TV)^{-1}W^TAV, & \hat{N}^{(k)} &= (W^TV)^{-1}W^TN^{(k)}V, \quad \text{for } k \in \{1,\dots,m\}, \\ \hat{B} &= (W^TV)^{-1}W^TB \qquad \text{and} & \hat{C} &= CV. \end{aligned}$$

Clearly, it can be seen that the quality of the reduced system (18.2) depends on the choice of the projection matrices. Several methods for linear systems have been extended to bilinear systems such as balanced truncation [7] and interpolation-based MOR [3, 5, 10, 13]. In this work, we mainly focus on a balanced truncation based MOR technique for bilinear systems. Balanced truncation for linear systems, $N_k = 0$ in (18.1), has been studied in, e.g., [2, 19], and relies on controllability and observability Gramians of the system. Later on, the balancing concept for general nonlinear systems has been extended in a series of papers; see, e.g., [14, 16, 22], where a new notion of controllability and observability energy functionals has been introduced. Although theoretically the balancing concept for nonlinear systems is appealing, it is seldom applicable from the computational perspective. This is due to the fact that the energy functionals are solutions of nonlinear Hamilton-Jacobi equations, which are extremely expensive to solve for large-scale systems.

Subsequently, the generalized Gramians for bilinear systems have been developed in regards to MOR; see, e.g. [1], which are the solutions of generalized Lyapunov equations of the form

$$AP + PA^T + \sum\nolimits_{k=1}^{m} N^{(k)}P\left(N^{(k)}\right)^T = -BB^T, \tag{18.3a}$$

$$A^TQ + QA + \sum\nolimits_{k=1}^{m} \left(N^{(k)}\right)^T QN^{(k)} = -C^TC, \tag{18.3b}$$

where A, $N^{(k)}$, B and C are as in (18.1). The connections between these Gramians and the energy functionals of bilinear systems have been studied in [7, 15]. Furthermore, the relations between the Gramians and the controllability/observability of bilinear systems have also been studied in [7]. However, the main bottleneck in using these Gramians in the MOR context is the computation of the Gramians, though recently there have been many advances in methods to determine the low-rank solutions of these generalized Lyapunov equations (18.3); see [6, 24].

This motivates us to investigate an alternative pair of Gramians for bilinear systems, which we call *Truncated* Gramians (TGrams). Regarding this, in Sect. 18.2 we recall balanced truncation for bilinear systems based on the Gramians (18.3). In Sect. 18.3, we propose TGrams for bilinear systems and investigate their connections with the controllability and observability of the bilinear systems. Moreover, we reveal the relation between these TGrams and energy functionals of the bilinear systems. Then, we discuss the advantages of considering these TGrams in the MOR context. Subsequently in Sect. 18.4, we provide a couple of numerical examples to illustrate the applicability of the TGrams for MOR of bilinear systems.

18.2 Background Work

In this section, we outline basic concepts of balanced truncation MOR. For this, let us consider a bilinear control system as in (18.1), then the controllability and observability of a state $x \in \mathbb{R}^n$ can be defined based on energy functionals as follows [22]:

$$E_c(x_0) = \inf_{\substack{u\in L^2(-\infty,0) \\ x(-\infty)=0,\ x(0)=x_0}} \frac{1}{2}\int_{-\infty}^{0} \|u(t)\|^2 dt, \qquad E_o(x_0) = \frac{1}{2}\int_{0}^{\infty} \|y(t)\|^2 dt, \tag{18.4}$$

respectively. The functional E_c is measured in terms of the minimal input energy required to steer the system from $x(-\infty) = 0$ to a desired state x_0 at time $t = 0$. If the state x_0 is uncontrollable, then it requires infinite energy; that means $E_c(x_0) = \infty$. On the other hand, the functional E_o characterizes the output energy determined by a particular initial state x_0 using the uncontrolled system. If the state

x_0 is unobservable, then it produces no output energy; $E_o(x_0) = 0$. In the linear case ($N_k = 0$), these energy functionals can be represented exactly by the Gramians of the linear system:

$$E_c(x) = \tfrac{1}{2}\langle x, P_l^{\#} x\rangle \quad \text{and} \quad E_o(x) = \tfrac{1}{2}\langle x, Q_l x\rangle,$$

where $\langle \cdot, \cdot \rangle$ represents the Euclidean inner product, $P_l^{\#}$ denotes the Moore-Penrose pseudo inverse of P_l, and P_l and Q_l are the unique and positive semidefinite solutions of the following Lyapunov equations:

$$AP_l + P_l A^T + BB^T = 0 \quad \text{and} \quad A^T Q_l + Q_l A + C^T C = 0, \tag{18.5}$$

respectively. In case of a nonlinear setting, the functionals E_c and E_o can be determined by solving Hamilton-Jacobi nonlinear PDEs, which are quite expensive to solve for large-scale settings. For more details on these PDEs, we refer to [22]. However, for MOR of bilinear systems, the Gramians, namely the controllability (P) and the observability (Q) Gramians, are defined as

$$\begin{aligned} P &= \sum_{k=1}^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} \bar{P}_k(t_1, \ldots, t_k) \bar{P}_k(t_1, \ldots, t_k)^T dt_1 \cdots dt_k, \\ Q &= \sum_{k=1}^{\infty} \int_0^{\infty} \cdots \int_0^{\infty} \bar{Q}_k(t_1, \ldots, t_k) \bar{Q}_k(t_1, \ldots, t_k)^T dt_1 \cdots dt_k, \end{aligned} \tag{18.6}$$

respectively, where

$$\begin{aligned} &\bar{P}_1(t_1) = e^{At_1} B, && \bar{P}_k(t_1, \ldots, t_k) = e^{At_k} \left[N^{(1)}, \ldots, N^{(k)} \right] \bar{P}_{k-1}, \\ &\bar{Q}_1(t_1) = e^{A^T t_1} C^T, && \bar{Q}_k(t_1, \ldots, t_k) = e^{A^T t_k} \left[\left(N^{(1)}\right)^T, \ldots, \left(N^{(k)}\right)^T \right] \bar{Q}_{k-1}. \end{aligned} \tag{18.7}$$

Then, the connections between these Gramians and Lyapunov equations are derived in [1]. Therein, it is shown that these Gramians satisfy the generalized Lyapunov equations stated in (18.3). Though energy functionals for bilinear system cannot be determined exactly in terms of the Gramians of the latter system, the Gramians provide a lower bound (locally) for the input (controllability) energy functional and an upper bound (locally) for the output (observability) energy functional as follows:

$$E_c(x) \geq \tfrac{1}{2}\langle x, P^{-1} x\rangle, \qquad \text{and} \qquad E_o(x) \leq \tfrac{1}{2}\langle x, Qx\rangle, \tag{18.8}$$

in a small open neighborhood of the origin [7, 15], where in (18.8) we assume that P and Q are positive definite.

However, in the general case with $P, Q \geq 0$ it is shown in [7] that if the desired state $x_0 \notin \operatorname{Im} P$, then $E_c(x_0) = \infty$, and similarly if an initial state x_0 belongs to

$\operatorname{Ker} Q$, then $E_o(x_0) = 0$. This shows that the states x_0 with $x_0 \in \operatorname{Ker} Q$ or $x_0 \notin \operatorname{Im} P$ do not play any role in the dynamics of the system; hence they can be removed. The main idea of balanced truncation lies in furthermore neglecting the almost uncontrollable and almost unobservable states (hard to control and hard to observe states).

In order to guarantee that hard to control and hard to observe states are truncated simultaneously, we need to find a transformation $x \mapsto T^{-1}x$, leading to a transformed bilinear system, whose controllability and observability Gramians are equal and diagonal, i.e.,

$$T^{-1}PT^{-T} = T^TQT = \Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n). \tag{18.9}$$

Analogous to the linear case (see, e.g., [2]), having the factorizations of $P = LL^T$ and $L^TQL = U\Sigma^2U^T$, one finds the corresponding transformation matrix in $T = LU\Sigma^{-\frac{1}{2}}$. Now, w.l.o.g. we consider the following bilinear system being a balanced bilinear system:

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \sum_{k=1}^{m} \begin{bmatrix} N_{11}^{(k)} & N_{12}^{(k)} \\ N_{21}^{(k)} & N_{22}^{(k)} \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} u_k(t) + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(t),$$
$$y(t) = \begin{bmatrix} C_1 & C_2 \end{bmatrix} \begin{bmatrix} x_1^T(t) & x_2^T(t) \end{bmatrix}^T,$$

with the controllability and observability Gramians equal to Σ :

$$\Sigma = \operatorname{diag}(\sigma_1, \sigma_2, \ldots, \sigma_n),$$

$\sigma_i \geq \sigma_{i+1}$ and $x_1(t) \in \mathbb{R}^r$ and $x_2(t) \in \mathbb{R}^{n-r}$. Fixing r such that $\sigma_r > \sigma_{r+1}$, we determine a reduced-order system of order r by neglecting x_2 as follows:

$$\begin{aligned} \dot{x}_1(t) &= A_{11}x_1(t) + \sum\nolimits_{k=1}^{m} N_{11}^{(k)} x_1(t) u_k(t) + B_1 u(t), \\ y_r(t) &= C_1 x_1(t). \end{aligned} \tag{18.10}$$

This provides a good local reduced-order system, but unlike in the linear case, it is not clear how to quantify the error, occurring due to x_2 being removed.

18.3 Truncated Gramians for Bilinear Systems

As discussed in the preceding section, we need to solve two generalized Lyapunov equations in order to compute reduced-order systems via balanced truncation. Solving these generalized Lyapunov equations is a numerical challenge for large-scale settings, although there have been many advancements in this direction in recent times; see, e.g. [6, 24]. In this section, we seek to determine TGrams for bilinear systems and discuss their advantages in the balancing-type MOR.

We define TGrams for bilinear systems by considering only the first two terms in the series in (18.6), which are dependent on the first two *kernels* of the Volterra series of the bilinear system, as follows:

$$P_T = \int_0^\infty \bar{P}_1(t_1)\bar{P}_1^T(t_1)dt + \int_0^\infty \int_0^\infty \bar{P}_2(t_1,t_2)\bar{P}_2^T(t_1,t_2)dt_1dt_2, \tag{18.11a}$$

$$Q_T = \int_0^\infty \bar{Q}_1(t_1)\bar{Q}_1^T(t_1)dt_1 + \int_0^\infty \int_0^\infty \bar{Q}_2(t_1,t_2)\bar{Q}_2^T(t_1,t_2)dt_1dt_2, \tag{18.11b}$$

where $\bar{P}_i$ and $\bar{Q}_i$ are defined in (18.7). Next, we establish the relations between these truncated Gramians and the solutions of Lyapunov equations.

Lemma 1 *Consider the bilinear system* (18.1) *and let P_T and Q_T be the truncated controllability and observability Gramians of the system as defined in* (18.11)*. Then, P_T and Q_T satisfy the following Lyapunov equations:*

$$AP_T + P_TA^T + \sum\nolimits_{k=1}^m N^{(k)}P_l\left(N^{(k)}\right)^T + BB^T = 0, \tag{18.12a}$$

$$A^TQ_T + Q_TA + \sum\nolimits_{k=1}^m \left(N^{(k)}\right)^T Q_lN^{(k)} + C^TC = 0, \tag{18.12b}$$

respectively, where P_l and Q_l are the Gramians of the linear systems as shown in (18.5)*.*

The above lemma can be proven in a similar way as done in [1, Thm. 1]. Therefore, due to space limitations, we omit the proof. Next, we compare the energy functionals of the bilinear system and the quadratic forms given by the TGrams. Before we state the corresponding lemma, we introduce the *homogeneous* bilinear system, which is used to characterize the observability energy in the system, as follows:

$$\begin{aligned}\dot{x}(t) &= Ax(t) + \sum\nolimits_{k=1}^m N^{(k)}x(t)u_k(t),\\ y(t) &= Cx(t), \quad x(0) = x_0.\end{aligned} \tag{18.13}$$

Lemma 2 *Given the bilinear system* (18.1)*, with an asymptotically stable A, being asymptotically reachable from* 0 *to any state x. Let $P, Q > 0$ and $P_T, Q_T > 0$ be the Gramians and* TGrams *of the system, respectively. Then, there exists a small neighborhood W of* 0*, where the following relations hold:*

$$E_c(x) \geq \tfrac{1}{2}x^TP_T^{-1}x \geq \tfrac{1}{2}x^TP^{-1}x \quad \textit{and} \quad x \in W(0). \tag{18.14}$$

Furthermore, there also exists a neighborhood $\hat{W}$ *of* 0*, where the following holds:*

$$E_o(x) \leq \tfrac{1}{2}x^T Q_T x \leq \tfrac{1}{2}x^T Q x \quad \textit{and} \quad x \in \hat{W}(0), \tag{18.15}$$

where E_c *and* E_o *are the energy functionals defined in* (18.4)*.*

Proof To prove (18.14), we follow the lines of reasoning in [7]. Let us assume that $x_0 \in \mathbb{R}^n$ is controlled by the input $u = u_{x_0} :]-\infty, 0] \to \mathbb{R}^m$, minimizing the input cost functional in the definition of $E_c(x_0)$. Using this input, we consider the homogeneous linear differential equation given by

$$\dot{\phi} = \left(A + \sum\nolimits_{k=1}^{m} N^{(k)} u_k(t)\right)\phi =: A_u(t)\phi(t), \tag{18.16}$$

whose fundamental solution is denoted by Φ_u. Thus, if we consider a time-varying system as $\dot{x} = A_u(t)x + Bu$, then its controllability Gramian can be given by

$$P_u = \int_{-\infty}^{0} \Phi_u(0,\tau)BB^T\Phi_u(0,\tau)^T d\tau. \tag{18.17}$$

The input u also controls the time-varying system from 0 to x_0; therefore we have

$$\|u\|_{L^2}^2 \geq x_0^T P_u^{\#} x_0, \tag{18.18}$$

where $P_u^{\#}$ denotes the Moore-Penrose pseudo inverse of P_u. Alternatively, one can also determine P_u as an observability Gramian as

$$P_u = \int_{0}^{\infty} \Psi_u(t,0)BB^T\Psi_u(t,0)^T dt, \tag{18.19}$$

where Ψ_u is the fundamental solution of the dual system satisfying

$$\dot{\Psi}_u = \left(A^T + \sum\nolimits_{k=1}^{m} \left(N^{(k)}\right)^T u_k(t)\right)\Psi_u, \quad \Psi_u(t,t) = I. \tag{18.20}$$

Note that we always choose x_0 in a small neighborhood W_0 of 0 such that only a small input is required to steer the systems from 0 to x_0, ensuring the asymptotic stability of $A_u(t)$. Hence, the matrix P_u is well-defined. Now, we define $\tilde{x}(t) = \Psi_u(t,0)x_0$, then we have

$$\begin{aligned} x_0^T P_T x_0 &= -\int_0^\infty \frac{d}{dt}\left(\tilde{x}(t)^T P_T \tilde{x}(t)\right) dt \\ &= -\int_0^\infty \tilde{x}(t)^T \Big(AP_T + \sum\nolimits_{k=1}^{m} N^{(k)} P_T u_k(-t) \\ &\qquad\qquad + P_T A^T + \sum\nolimits_{k=1}^{m} P_T \left(N^{(k)}\right)^T u_k(-t)\Big)\tilde{x}(t)dt \end{aligned}$$

$$= -\int_0^\infty \tilde{x}(t)^T \left(AP_T + P_T A^T + \sum\nolimits_{k=1}^m N^{(k)} P_l \left(N^{(k)}\right)^T\right) \tilde{x}(t)dt + \int\limits_0^\infty \tilde{x}(t)^T$$

$$\times \sum_{k=1}^m \left(N^{(k)} P_l \left(N^{(k)}\right)^T - N^{(k)} P_T u_k(-t) - P_T \left(N^{(k)}\right)^T u_k(-t)\right) \tilde{x}(t)dt.$$

Thus, we get

$$\begin{aligned} -\int_0^\infty \tilde{x}(t)^T \left(AP_T + P_T A^T + \sum\nolimits_{k=1}^m N^{(k)} P_l \left(N^{(k)}\right)^T\right) \tilde{x}(t)dt &= \int_0^\infty \tilde{x}(t)^T BB^T \tilde{x}(t)dt \\ &= x_0^T P_u x_0. \end{aligned}$$

Moreover, if

$$\int\limits_0^\infty \tilde{x}(t)^T \sum_{k=1}^m \left(N^{(k)} P_l \left(N^{(k)}\right)^T - N^{(k)} P_T u_k(-t) - P_T \left(N^{(k)}\right)^T u_k(-t)\right) \tilde{x}(t)dt \geq 0, \tag{18.21}$$

then we have $x_0^T P_T x_0 \geq x_0^T P_u x_0$. Furthermore, if we assume that the reachable state x_0 lies in a sufficiently small ball W in the neighborhood of 0 and $W(0) \subseteq W_0(0)$, then x_0 is reached with a sufficiently small input u, guaranteeing that the condition (18.21) is satisfied for all states $x_0 \in W(0)$. Hence, we obtain

$$x_0^T P_T^{-1} x_0 \leq x_0^T P_u^{-1} x_0, \quad \text{where} \quad x_0 \in W(0).$$

Furthermore, if the controllability Gramian P, which is the solution of (18.3a), is determined as a series [12], then it is easy to conclude that $P \geq P_T \geq 0$. That means, $x_0^T P^{-1} x_0 \leq x_0^T P_T^{-1} x_0$. Thus, we have $x_0^T P^{-1} x_0 \leq x_0^T P_T^{-1} x_0 \leq x_0^T P_u^{-1} x_0$, where $x_0 \in W(0)$.

Furthermore, along the lines of the proof [15, Thm 3.3], we can prove that

$$E_o(x_0, u) - \tfrac{1}{2} x_0^T Q_T x_0 = \int_0^\infty x(t)^T \mathscr{R}(t) x(t)dt,$$

where $\mathscr{R}(t) = \sum\limits_{k=1}^m \left(Q_T N^{(k)} u_k(t) - \frac{1}{2}\left(N^{(k)}\right)^T Q_l N^{(k)}\right)$. For sufficiently small input u, it can be seen that $\mathscr{R}(t)$ is a negative semidefinite matrix. Hence, we get

$$E_o(x_0, u) - \tfrac{1}{2} x_0^T Q_T x_0 \leq 0 \quad \Rightarrow \quad E_o(x_0, u) \leq \tfrac{1}{2} x_0^T Q_T x_0.$$

Moreover, if the observability Gramian is determined as a series with positive semidefinite summands, then it can also be seen that $Q \geq Q_T$; hence

$$E_o(x_0, u) \leq \tfrac{1}{2} x_0^T Q_T x_0 \leq \tfrac{1}{2} x_0^T Q x_0.$$

This concludes the proof. □

To illustrate the relation between energy functionals, Gramians and TGrams of bilinear systems, we consider the same scalar example considered in [15].

Example 1 Consider a scalar example (a, b, c, η). We assume $a < 0$, $\eta^2 + 2a < 0$ and $bc \neq 0$ to ensure the existence of $P, Q > 0$. The energy functionals of the system can be determined by solving the corresponding nonlinear PDEs [15], which are:

$$E_c(x) = \frac{2a}{\eta^2}\left[\frac{\eta x}{\eta x + b} + \log\left(\frac{b}{\eta x + b}\right)\right] \quad \text{and} \quad E_o(x) = -\frac{1}{2}\left(\frac{c^2}{2a}\right)x^2.$$

The approximations of the energy functionals using the Gramians are:

$$E_c^{(G)}(x) = \frac{1}{2}\left(\frac{\eta^2 + 2a}{-b^2}\right)x^2 \quad \text{and} \quad E_o^{(G)}(x) = \frac{1}{2}\left(\frac{-c^2}{\eta^2 + 2a}\right)x^2.$$

The approximations of the energy functionals using TGrams are:

$$E_c^{(T)}(x) = a\left(-b^2 + \frac{\eta^2 b^2}{2a}\right)^{-1} x^2 \quad \text{and} \quad E_o^{(T)}(x) = \frac{1}{4a}\left(-c^2 + \frac{\eta^2 c^2}{2a}\right)x^2.$$

The comparison of these quantities by taking numerical values for $-a = b = c = \eta = 1$ is illustrated in Fig. 18.1.

Next, we recall the discussion in [7] about definiteness of Gramians and controllability/observability of the bilinear systems. Following this discussion, we also show how controllability/observability of the bilinear systems are related to the TGrams.

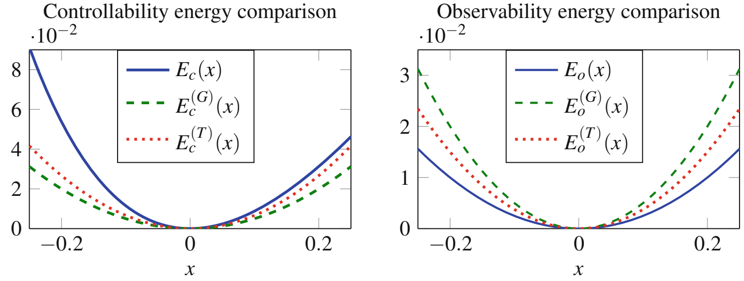


Fig. 18.1 The figure shows the comparison of the energy functionals of the system and their approximations via Gramians and TGrams as stated in Lemma 2

Theorem 1

(a) Consider the bilinear system (18.1) *and define its truncated controllability Gramian P_T as in* (18.11a)*. If the final state $x_0 \notin \operatorname{Im} P_T$, then $E_c(x_0) = \infty$.*

(b) Consider the homogeneous bilinear system (18.13) *and assume that the truncated observability Gramian Q_T is defined as in* (18.11b)*. If the initial condition $x_0 \in \operatorname{Ker} Q_T$, then the output $y(t)$ is zero for all $t \geq 0$, i.e. $E_o(x_0) = 0$.*

Proof The above theorem can be proven in a similar manner as [7, Thm. 3.1] using one of the important properties of positive semidefinite matrices. It is that the null space of the matrix $\mathscr{C}$, which is the sum of two positive semidefinite matrices $\mathscr{A}$ and $\mathscr{B}$, is the intersection of the null space of $\mathscr{A}$ and $\mathscr{B}$. In other words, if the vector v belongs to the null space of $\mathscr{C}$, then $\mathscr{A}v = 0$ and $\mathscr{B}v = 0$ as well. However, we skip a detailed proof due to the limited space. □

From Lemma 2 and Theorem 1, it is clear that the TGrams for bilinear systems can also be used to determine the states that absorb a lot of energy, and still produce very little output energy. However, there are several advantages of considering the TGrams over the Gramians for bilinear systems. Firstly, TGrams approximate the energy functionals of the bilinear systems more accurately (at least locally) as proven in Lemma 2 and illustrated in Example 1. Secondly, in order to compute TGrams, we require the solutions of four conventional Lyapunov equations, whereas the Gramians require the solutions of the generalized Lyapunov equations (18.3), which are indeed much more computationally cumbersome. Lastly, TGrams are of smaller rank as compared to Gramians; i.e. $P \geq P_T$ and $Q \geq Q_T$. This indicates that $\sigma_i(P \cdot Q) \geq \sigma_i(P_T \cdot Q_T)$, where $\sigma_i(\cdot)$ denotes the i-th largest eigenvalue of the matrix. This can be shown using Weyl's inequality [25]. Hence, if one chooses to truncate at machine precision, then the reduced system based on TGrams is probably to be of a small order; however, the relative decay of the Hankel singular values $\left(\sqrt{\sigma_i(P_T \cdot Q_T)}\right)$ so far lacks any analysis.

18.4 Numerical Results

In this section, we illustrate the efficiency of the reduced-order systems obtained via the proposed TGrams for the bilinear system and compare it with that of the full Gramians [7]. We denote the Gramians for the bilinear system by SGrams (*standard* Gramians) from now on. In order to determine the low-rank factors of the Gramians for bilinear systems, we employ the most recently proposed algorithm in [24], which utilizes many of the properties of inexact solutions and uses the extended Krylov subspace method (EKSM) to solve the conventional Lyapunov equation up to a desired accuracy. To determine the low-rank factors of the linear Lyapunov equation, we also utilize EKSM to be in the same line. All the simulations were carried out in MATLAB® version 7.11.0(R2010b) on a board with 4 Intel®

Xeon®E7-8837 CPUs with a 2.67-GHz clock speed, 8 Cores each and 1TB of total RAM.

18.4.1 Burgers' Equation

We consider a viscous Burgers' equation, which is one of the standard test examples for bilinear systems; see, e.g. [10]. Therein, one can also find the governing equation, boundary conditions and initial condition of the system. As shown in [10], a spatial semi-discretization of the governing equation using k equidistant nodes leads to an ODE system with quadratic nonlinearity. However, the quadratic nonlinear system can be approximated using Carleman bilinearization; see, e.g., [21]. The dimension of the approximated bilinearized system is $n = k+k^2$. We set the viscosity $\mu = 0.1$ and $k = 40$, and choose the observation vector C such that it yields an average value for the variable v in the spatial domain. The bilinearized system is not an $\mathscr{H}_2$ system, which can be checked by looking at the eigenvalues of the matrix $\mathscr{X} := (I \otimes A + A \otimes I + N \otimes N)$. If $\sigma(\mathscr{X}) \not\subset \mathbb{C}^-$, then the series determining its controllability Gramians diverges. To overcome this issue, we choose a scaling factor γ, which multiplies with the matrices B and N_k, and the input $u(t)$ is scaled by $\frac{1}{\gamma}$. For this example, we set $\gamma = 0.1$, ensuring $\sigma(\mathscr{X}) \subset \mathbb{C}^-$.

We determine reduced systems of orders $r = 5$ and $r = 10$ using SGrams and TGrams, and compare the quality of the reduced-order systems by using two arbitrary control inputs as shown in Fig. 18.2. More importantly, we also show the CPU-time to determine the low-rank factors of SGrams and TGrams in the same figure.

Figure 18.2 shows that computing TGrams is much cheaper than SGrams. Moreover, we observe that the reduced systems based on TGrams are very much competitive to those of SGrams for both control inputs and both orders in Example 18.4.1.

18.4.2 Electricity Cable Impacted by Wind

Below, we discuss an example studied in [20]. Therein, a damped wave equation with Lévy noise is considered, which is transformed into a first order stochastic PDE (SPDE) and then discretized in space. The governing equation, which models the lateral displacement of an electricity cable impacted by wind, is

$$\frac{\partial^2}{\partial t^2}X(t,z) + 2\frac{\partial}{\partial t}X(t,z) = \frac{\partial^2}{\partial z^2}X(t,z) + \mathrm{e}^{-(z-\frac{\pi}{2})^2}\,u(t) + 2\,\mathrm{e}^{-(z-\frac{\pi}{2})^2}X(t-,z)\frac{\partial M(t)}{\partial t}$$

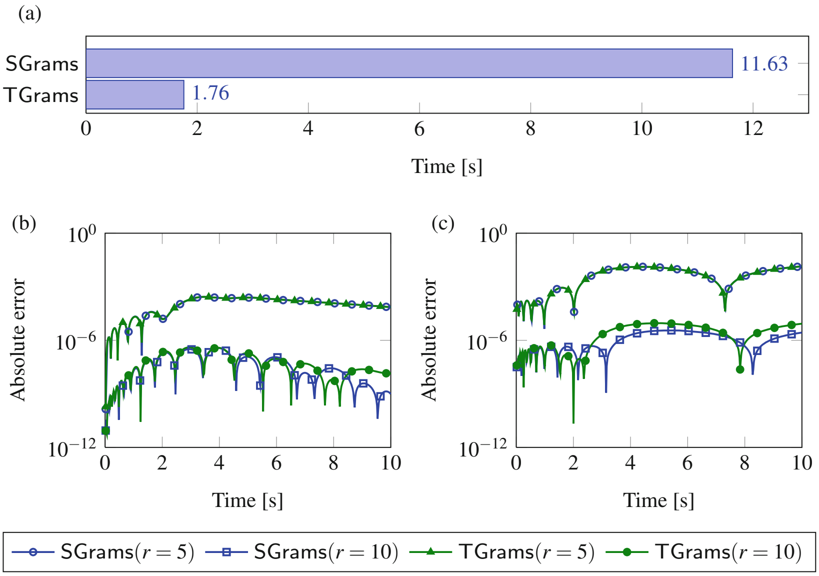


Fig. 18.2 Comparisons of CPU-time and time-domain responses of the original and reduced-order systems for two different orders and for two inputs for the Burgers' equation example. (**a**) Comparison of CPU-time to compute SGrams and TGrams for Example 18.4.1. (**b**) For an input $u(t) = t \cdot e^{-t} \cdot \sin(\pi t)$. (**c**) For an input $u(t) = t \cdot e^{-t} + 1$

s for $t, z \in [0, \pi]$, where M is a scalar, square integrable Lévy process with mean zero. The boundary and initial conditions are:

$$X(t,0) = X(t,\pi) = 0 \quad \text{and} \quad X(0,z) = 0, \left.\frac{\partial}{\partial t}X(t,z)\right|_{t=0} \equiv 0.$$

An approximation for the position of the middle of the cable represents the output

$$Y(t) = \frac{1}{2\varepsilon}\int_{\frac{\pi}{2}-\varepsilon}^{\frac{\pi}{2}+\varepsilon} X(t,z)dz, \quad \varepsilon > 0.$$

Following [20], a semi-discretized version of the above SPDE has the following form with $x(0) = 0$ and $t \in [0, \pi]$:

$$dx(t) = [Ax(t) + Bu(t)]\,dt + Nx(s-)dM(s), \quad y(t) = Cx(t). \tag{18.22}$$

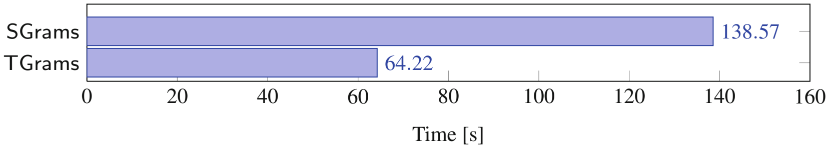


Fig. 18.3 Comparison of CPU-time to compute SGrams and TGrams for Example 18.4.2

Here, A, $N \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$, $x(t-) := \lim_{s\uparrow t} x(s)$ and y is the corresponding output. We, moreover, assume that the adapted control satisfies $\|u\|^2_{\mathscr{L}^2_T} := \mathbb{E}\int_0^T \|u(t)\|^2_{\mathbb{R}^m}\, dt < \infty$. For more details, we refer to [20].

In contrast to [20], we fix a different noise process, which allows the wind to come from two directions instead of just one. The noise term we choose is represented by a compound Poisson process $M(t) = \sum_{i=1}^{N(t)} Z_i$ with $(N(t))_{t\in[0,\pi]}$ being a Poisson process with parameter 1. Furthermore, $Z_1, Z_2, \ldots$ are independent uniformly distributed random variables with $Z_i \sim \mathscr{U}\left(-\sqrt{3}, \sqrt{3}\right)$, which are also independent of $(N(t))_{t\in[0,\pi]}$. This choice implies $\mathbb{E}\left[M(t)\right] = 0$ and $\mathbb{E}\left[M^2(1)\right] = 1$. BT for such an Ito type SDE (18.22) with the particular choice of M is also based on Gramians, which fulfill equations (18.3) with $m = 1$ and $N := N^{(1)}$. We fix the dimension of (18.22) to $n = 1000$ and set $u(t) = e^{w(t)}\sin(t)$, and then run several numerical experiments.

We apply BT based on SGrams as described in [9] and compute the reduced systems of order $r = 3$ and $r = 6$. Similarly, we determine the reduced systems of the same orders using TGrams. Next, we discuss the quality of these derived reduced systems and computational cost to determine the low-rank factors of SGrams and TGrams. In Fig. 18.3, we see that the TGrams are computationally much cheaper as compared to the SGrams.

For the $r = 3$ case, clearly the reduced system based on TGrams outperforms the one based on the SGrams for all three trajectories (see Fig. 18.4a). This is also true for the mean deviation as shown in the Fig. 18.4c (left). For the $r = 6$ case, it is not that obvious anymore. The reduced system obtained by SGrams seems to be marginally more accurate, but still both methods result in very competitive reduced-order systems, see Fig. 18.4b, c (right).

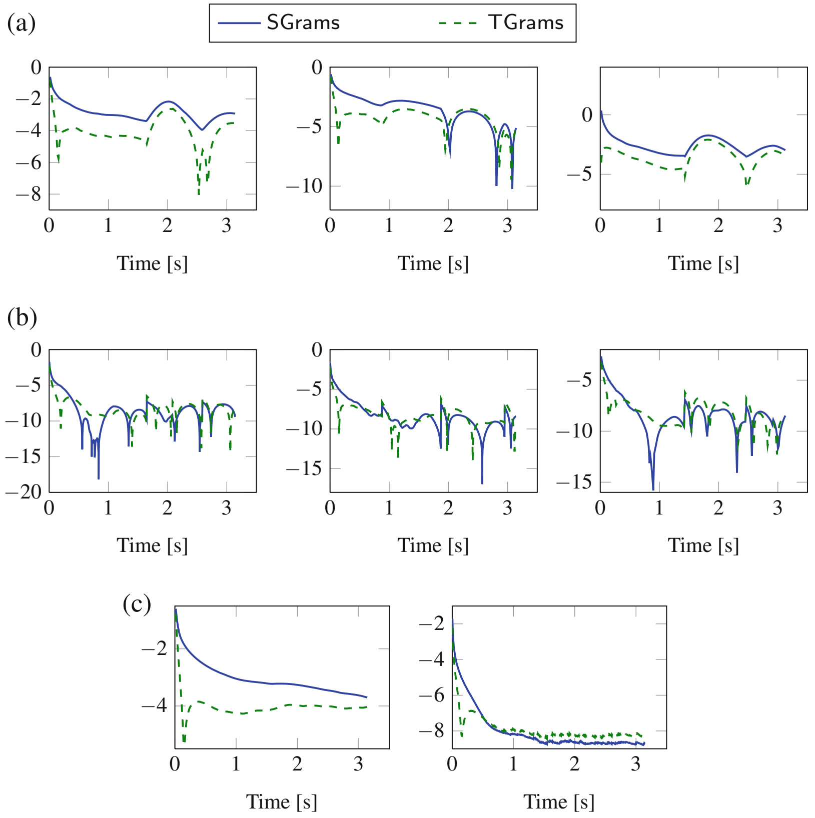


Fig. 18.4 Comparison of reduced-order systems for $u(t) = e^{w(t)}\sin(t)$. (**a**) $\ln\left(\frac{|y(\omega,t)-y_r(\omega,t)|}{|y(\omega,t)|}\right)$ with reduced order dimension $r = 3$. (**b**) $\ln\left(\frac{|y(\omega,t)-y_r(\omega,t)|}{|y(\omega,t)|}\right)$ with reduced order dimension $r = 6$. (**c**) $\ln\left(\frac{\mathbb{E}|y(t)-y_r(t)|}{\mathbb{E}|y(t)|}\right)$, where $r = 3$ (*left*), 6 (*right*)

18.5 Conclusions

In this paper, we have proposed truncated Gramians for bilinear systems. These allow us to find the states, which are both hard to control and hard to observe, like the Gramians for bilinear systems. We have also shown that the truncated Gramians approximate the energy functionals of bilinear systems better (at least locally) as compared to the Gramians of the latter systems. We have presented how controllability and observability of bilinear systems are related to the truncated Gramians. Moreover, we have discussed advantages of the truncated Gramians in the model reduction context. In the end, we have demonstrated the efficiency of
]

the proposed truncated Gramians in model reduction by means of two numerical examples.

Acknowledgements The authors thank Dr. Stephen D. Shank for providing the MATLAB implementation to compute the low-rank solutions of the generalized Lyapunov equations.

References

1. Al-baiyat, S. A, Bettayeb, M.: A new model reduction scheme for k-power bilinear systems. In: Proceedings of 32nd IEEE CDC, pp. 22–27. IEEE, Piscataway (1993)
2. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. SIAM, Philadelphia (2005)
3. Bai, Z., Skoogh, D.: A projection method for model reduction of bilinear dynamical systems. Linear Algebra Appl. **415**(2–3), 406–425 (2006)
4. Benner, P., Breiten, T.: On $\mathscr{H}_2$-model reduction of linear parameter-varying systems. In: Proceedings of Applied Mathematics and Mechanics, vol. 11, pp. 805–806 (2011)
5. Benner, P., Breiten, T.: Interpolation-based $\mathscr{H}_2$-model reduction of bilinear control systems. SIAM J. Matrix Anal. Appl. **33**(3), 859–885 (2012)
6. Benner, P., Breiten, T.: Low rank methods for a class of generalized Lyapunov equations and related issues. Numer. Math. **124**(3), 441–470 (2013)
7. Benner, P., Damm, T.: Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems. SIAM J. Cont. Optim. **49**(2), 686–711 (2011)
8. Benner, P., Mehrmann, V., Sorensen, D.C.: Dimension Reduction of Large-Scale Systems. Lecture Notes in Computational Science and Engineering, vol. 45. Springer, Berlin/Heidelberg (2005)
9. Benner, P., Redmann, M.: Model reduction for stochastic systems. Stoch. PDE: Anal. Comp. **3**(3), 291–338 (2015)
10. Breiten, T., Damm, T.: Krylov subspace methods for model order reduction of bilinear control systems. Syst. Control Lett. **59**(10), 443–450 (2010)
11. Bruni, C., DiPillo, G., Koch, G.: On the mathematical models of bilinear systems. Automatica **2**(1), 11–26 (1971)
12. Damm, T.: Direct methods and ADI-preconditioned Krylov subspace methods for generalized Lyapunov equations. Numer. Linear Algebra Appl. **15**(9), 853–871 (2008)
13. Flagg, G., Gugercin, S.: Multipoint Volterra series interpolation and $\mathscr{H}_2$ optimal model reduction of bilinear systems. SIAM. J. Matrix Anal. Appl. **36**(2), 549–579 (2015)
14. Fujimoto, K., Scherpen, J.M.A.: Balanced realization and model order reduction for nonlinear systems based on singular value analysis. SIAM J. Cont. Optim. **48**(7), 4591–4623 (2010)
15. Gray, W.S., Mesko, J.: Energy functions and algebraic Gramians for bilinear systems. In: Preprints of the 4th IFAC Nonlinear Control Systems Design Symposium, Enschede, The Netherlands, pp. 103–108 (1998)
16. Gray, W.S., Scherpen, J.M.A.: On the nonuniqueness of singular value functions and balanced nonlinear realizations. Syst. Control Lett. **44**(3), 219–232 (2001)
17. Hartmann, C., Schäfer-Bung, B., Thons-Zueva, A.: Balanced averaging of bilinear systems with applications to stochastic control. SIAM J. Cont. Optim. **51**(3), 2356–2378 (2013)
18. Mohler, R.R.: Bilinear Control Processes. Academic Press, New York (1973)
19. Moore, B.C.: Principal component analysis in linear systems: controllability, observability, and model reduction. IEEE Trans. Autom. Control **AC-26**(1), 17–32 (1981)
20. Redmann, M., Benner, P.: Approximation and model order reduction for second order systems with Levy-noise. In: Dynamical Systems, Differential Equations and Applications AIMS Proceedings, pp. 945–953, 2015

21. Rugh, W.J.: Nonlinear System Theory. The Johns Hopkins University Press, Baltimore (1981)
22. Scherpen, J.M.A.: Balancing for nonlinear systems. Syst. Control Lett. **21**, 143–153 (1993)
23. Schilders, W.H.A., van der Vorst, H. A., Rommes, J.: Model Order Reduction: Theory, Research Aspects and Applications. Springer, Berlin/Heidelberg (2008)
24. Shank, S.D., Simoncini, V., Szyld, D.B.: Efficient low-rank solution of generalized Lyapunov equations. Numer. Math. **134**(2), 327–342 (2016)
25. Weyl, H.: Das asymptotische Verteilungsgesetz der Eigenwerte linearer partieller Differentialgleichungen (mit einer Anwendung auf die Theorie der Hohlraumstrahlung). Math. Ann. **71**(4), 441–479 (1912)

Chapter 19
Leveraging Sparsity and Compressive Sensing for Reduced Order Modeling

J. Nathan Kutz, Syuzanna Sargsyan, and Steven L. Brunton

Abstract Sparsity can be leveraged with dimensionality-reduction techniques to characterize and model parametrized nonlinear dynamical systems. Sparsity is used for both sparse representation, via proper orthogonal decomposition (POD) modes in different dynamical regimes, and by compressive sensing, which provides the mathematical architecture for robust classification of POD subspaces. The method relies on constructing POD libraries in order to characterize the dominant, low-rank coherent structures. Using a greedy sampling algorithm, such as gappy POD and one of its many variants, an accurate Galerkin-POD projection approximating the nonlinear terms from a sparse number of grid points can be constructed. The selected grid points for sampling, if chosen well, can be shown to be effective sensing/measurement locations for classifying the underlying dynamics and reconstruction of the nonlinear dynamical system. The use of sparse sampling for interpolating nonlinearities and classification of appropriate POD modes facilitates a family of local reduced-order models for each physical regime, rather than a higher-order global model. We demonstrate the sparse sampling and classification method on the canonical problem of flow around a cylinder. The method allows for a robust mathematical framework for robustly selecting POD modes from a library, accurately constructing the full state space, and generating a Galerkin-POD projection for simulating the nonlinear dynamical system.

19.1 Introduction

Reduced-order models (ROMs) are of growing importance in scientific applications and computing as they help reduce the computational complexity and time needed to solve large-scale, engineering systems [7, 41]. For many complex systems of interest, simulations reveal that the dynamics of the system are *sparse* in the sense

J.N. Kutz (✉) • S. Sargsyan
Department of Applied Mathematics, University of Washington, Seattle, WA 98195-3925, USA
e-mail: kutz@uw.edu; ssuzie@uw.edu

S.L. Brunton
Department of Mechanical Engineering, University of Washington, Seattle, WA 98195, USA
e-mail: sbrunton@uw.edu

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_19

that a relatively small subset of the full space is needed to describe the evolution of the system. Thus solutions can often be approximated through dimensionality reduction methods where if n is the dimension of the original system and r is the dimension of the subspace (or slow-manifold) where the sparse dynamics is embedded, then $r \ll n$. Familiar examples include large-scale patterns of atmospheric variability, fluid dynamics and/or network simulations of neurosensory systems, which reveal that many variables are correlated. As a result, the large-scale dynamics may be compressed, or encoded, in a much smaller space than the full space using, for instance, the proper orthogonal decomposition (POD). The dynamics of the encoded subspace may be deduced from a modal (Galerkin) projection, observations of the system and/or constraints from physical laws (e.g. conservation of energy). In this manuscript, we show that the role of sparsity goes far beyond simply constructing low-dimensional embeddings. Specifically, we show that *compressive sensing* and *sparse representation* can be used in a highly effective manner for model reduction for both reconstruction and classification of parametrized systems.

The effective use of ROMs relies on constructing appropriate low-dimensional subspaces for projecting the dynamics along with sparse interpolation methods for evaluating nonlinearities and inner products. As such, there are three key parameters for characterizing ROM architectures:

$$\begin{aligned} n &- \text{dimension of the original system} \\ r &- \text{target rank of low-dimensional system} \\ m &- \text{number of measurements for interpolation} \end{aligned}$$

where $n \gg r, m$ and $m > r$ but with $m \sim r$. The interpolation measurements are similar to, but greater than, the rank of the low-dimensional system.

For parametrized nonlinear dynamical systems, the parameter μ generally guarantees the existence of a number of distinct dynamical regimes that can each be represented by a r-rank dynamical system through Galerkin-POD projection. We capitalize on this fact by building libraries of POD modes for different dynamical regimes associated with μ, which is similar to modern machine learning methods for clustering and classification of distinct features of data [8, 36] Thus a suite of local ROMs are constructed to avoid the well-known numerical instabilities generated from a global model encompassing many parameter regimes [7, 17]. Once the POD modes are constructed, then the same sparse sampling strategies using m measurements (e.g. gappy POD [7, 41]) that enable efficient evaluation of nonlinear terms can be used to first identify (classify) the current dynamical regime of interest before constructing an appropriate r-rank ROM model. The rank r of the model selected is dependent on the specific dynamical regime whereas the m sparse sampling (interpolation) points are chosen to be effective across all POD library modes. We demonstrate a method for constructing the library modes, selecting interpolation points, and constructing ROMs from appropriate parameter regimes. Sparsity techniques are central to integrating the overall mathematical framework.

19.2 Nonlinear Model Order Reduction and POD

The success of nonlinear model order reduction is largely dependent upon two key innovations: (1) the well-known POD-Galerkin method [30], which is used for projecting the high-dimensional nonlinear dynamics to a low-dimensional subspace in a principled way, and (2) sparse sampling of the state space for interpolating the nonlinear terms required for the subspace projection. Thus sparsity is already established as a critically enabling mathematical framework for model reduction through methods such as gappy POD and its variants [16, 28, 45, 46]. Indeed, efficiently managing the computation of the nonlinearity was recognized early on in the ROMs community, and a variety of techniques where proposed to accomplish the task. Perhaps the first innovation in sparse sampling with POD modes was the technique proposed by Everson and Sirovich for which the gappy POD moniker was derived [28]. In their sparse sampling scheme, random measurements were used to perform reconstruction tasks of inner products. Principled selection of the interpolation points, through the gappy POD infrastructure [16, 28, 45, 46] or missing point (best points) estimation (MPE) [4, 37], were quickly incorporated into ROMs to improve performance. More recently, the empirical interpolation method (EIM) [6] and its most successful variant, the POD-tailored discrete empirical interpolation method (DEIM) [18], have provided a greedy algorithm that allows for nearly optimal reconstructions of nonlinear terms of the original high-dimensional system. The DEIM approach combines projection with interpolation. Specifically, the DEIM uses selected interpolation indices to specify an interpolation-based projection for a nearly optimal ℓ_2 subspace approximating the nonlinearity.

Consider a parametrized, high-dimensional system of nonlinear differential equations that can arise, for example, from the finite-difference discretization of a partial differential equation:

$$\frac{d\mathbf{u}(t)}{dt} = L\mathbf{u}(t) + N(\mathbf{u}(t), \mu), \tag{19.1}$$

where $\mathbf{u}(t) = [u_1(t)\ u_2(t)\ \cdots\ u_n(t)]^T \in \mathbb{R}^n$ and $n \gg 1$. Typically, $u_j(t) = u(x_j, t)$ is the value of the field of interest discretized at the spatial location x_j. The linear part of the dynamics is given by $L \in \mathbb{R}^{n\times n}$ and the nonlinear terms are in the vector $N(\mathbf{u}(t)) = [N_1(\mathbf{u}(t), \mu) \quad N_2(\mathbf{u}(t), \mu) \quad \cdots \quad N_n(\mathbf{u}(t)), \mu]^T \in \mathbb{R}^n$. The nonlinear function is evaluated component-wise at the n spatial grid points used for discretization. Note that we have assumed, without loss of generality, that the parametric dependence μ is in the nonlinear term.

For achieving high-accuracy solutions, n is typically required to be very large, thus making the computation of the solution expensive and/or intractable. The POD-Galerkin method is a principled dimensionality-reduction scheme that approximates the function $\mathbf{u}(t)$ with rank-r-optimal basis functions where $r \ll n$. These optimal basis functions are computed from a singular value decomposition of a series of temporal snapshots of the nonlinear dynamical system. Specifically, suppose snapshots of the state, $\mathbf{u}(t_j)$ with $j = 1, 2, \cdots, p$, are collected. The snapshot matrix $\mathbf{X} = [\mathbf{u}(t_1)\ \mathbf{u}(t_2)\ \cdots\ \mathbf{u}(t_p)] \in \mathbb{R}^{n\times p}$ is constructed and the SVD of $\mathbf{X}$ is computed: $\mathbf{X} = \boldsymbol{\Psi}\boldsymbol{\Sigma}\mathbf{W}^*$. The r-dimensional basis for optimally approximating $\mathbf{u}(t)$ is given by the first r columns of matrix $\boldsymbol{\Psi}$, denoted by $\boldsymbol{\Psi}_r$. The POD-Galerkin approximation is

$$\mathbf{u}(t) \approx \boldsymbol{\Psi}_r\mathbf{a}(t) \tag{19.2}$$

where $\mathbf{a}(t) \in \mathbb{R}^r$ is the time-dependent coefficient vector and $r \ll n$. Plugging this modal expansion into the governing equation (19.1) and applying orthogonality (multiplying by $\boldsymbol{\Psi}_r^T$) gives the dimensionally reduced evolution

$$\frac{d\mathbf{a}(t)}{dt} = \boldsymbol{\Psi}_r^T L\boldsymbol{\Psi}_r\mathbf{a}(t) + \boldsymbol{\Psi}_r^T N(\boldsymbol{\Psi}_r\mathbf{a}(t), \mu). \tag{19.3}$$

By solving this system of much smaller dimension, the solution of a high-dimensional nonlinear dynamical system can be approximated. Of critical importance is evaluating the nonlinear terms in an efficient way using the gappy POD or DEIM mathematical architecture. Otherwise, the evaluation of the nonlinear terms still requires calculation of functions and inner products with the original dimension n. In certain cases, such as the quadratic nonlinearity of Navier-Stokes, the nonlinear terms can be computed once in an off-line manner. However, parametrized systems generally require repeated evaluation of the nonlinear terms as the POD modes change with μ.

19.3 Sparse Sampling for ROMs

The POD method aims to exploit the underlying low-dimensional dynamics observed in many high-dimensional computations. Although POD reductions are common for dimensionality reduction [30], the key to producing a viable ROM is to evaluate the nonlinear terms in (19.3). Specifically, a major shortcoming of the POD-Galerkin method can be generally due to the evaluation of the nonlinear term $N(\boldsymbol{\Psi}_r\mathbf{a}(t), \mu)$. To avoid this difficulty, sparse sampling (gappy POD, EIM, DEIM) approximates $\mathbf{N} = N(\boldsymbol{\Psi}_r\mathbf{a}(t), \mu)$ through projection and interpolation instead of evaluating it directly. A considerable reduction in complexity is achieved by sparse sampling because evaluating the approximate nonlinear term does not require a prolongation of the reduced state variables back to the original high dimensional state approximation required to evaluate the nonlinearity in the POD approximation. The method therefore improves the efficiency of the POD approximation and achieves a complexity reduction of the nonlinear term with a complexity proportional to the number of reduced variables. Sparse sampling constructs these specially selected interpolation indices that specify an interpolation-based projection to provide a nearly ℓ_2 optimal subspace approximation to the nonlinear term without the expense of orthogonal projection [18].

In particular, only $m \ll n$ measurements are required for reconstruction, allowing us to define the sparse representation variable $\tilde{\mathbf{u}} \in \mathbb{R}^m$

$$\tilde{\mathbf{u}} = \mathbf{P}\mathbf{u} \tag{19.4}$$

where the measurement matrix $\mathbf{P} \in \mathbb{R}^{m \times n}$ specifies m measurement locations of the full state $\mathbf{u} \in \mathbb{R}^n$. As an example, the measurement matrix might take the form

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & \cdots & & & & & \cdots 0 \\ 0 \cdots & 0 & 1 & 0 & \cdots & & & \cdots 0 \\ 0 \cdots & & & \cdots & 0 & 1 & 0 & \cdots 0 \\ \vdots & 0 & & \cdots & 0 & 0 & 1 & \cdots \vdots \\ 0 \cdots & & & \cdots & 0 & 0 & 0 & \cdots 1 \end{bmatrix} \tag{19.5}$$

where measurement locations take on the value of unity and the matrix elements are zero elsewhere. The matrix $\mathbf{P}$ defines a projection onto an m-dimensional measurement space $\tilde{\mathbf{u}}$ that is used to approximate $\mathbf{u}$.

The insight and observation of (19.4) forms the basis of the *Gappy POD* method introduced by Everson and Sirovich [28]. In particular, one can use a small number of measurements, or gappy data, to reconstruct the full state of the system. In doing so, we can overcome the complexity of evaluating higher-order nonlinear terms in the POD reduction.

The measurement matrix $\mathbf{P}$ allows for an approximation of the state vector $\mathbf{u}$ from m measurements. The approximation is given by using (19.4) with (19.11):

$$\tilde{\mathbf{u}} \approx \mathbf{P} \sum_{j=1}^{m} \tilde{a}_j \psi_j \tag{19.6}$$

where the coefficients $\tilde{a}_j$ minimize the error in approximation: $\|\tilde{\mathbf{u}} - \mathbf{P}\mathbf{u}\|$. The challenge is now how to determine the $\tilde{a}_j$ given the fact that taking inner products of (19.6) can no longer be performed. Specifically, the vector $\tilde{\mathbf{u}}$ is of length m whereas the POD modes are of length n, i.e. the inner product requires information from the full range of $\mathbf{x}$, the underlying discretized spatial variable which is of length n. Thus the modes $\psi_j(x)$ are in general not orthogonal over the m-dimensional support of $\tilde{\mathbf{u}}$. The support will be denoted as $s[\tilde{\mathbf{u}}]$. More precisely, orthogonality must be considered on the full range versus the support space. Thus the following two relationships hold

$$M_{jk} = \left(\psi_j, \psi_k\right) = \delta_{jk} \tag{19.7a}$$

$$M_{jk} = \left(\psi_j, \psi_k\right)_{s[\tilde{\mathbf{u}}]} \neq 0 \quad \text{for all}\, j, k \tag{19.7b}$$

where M_{jk} are the entries of the Hermitian matrix $\mathbf{M}$ and δ_{jk} is the Kronecker delta function. The fact that the POD modes are not orthogonal on the support $s[\tilde{\mathbf{u}}]$ leads us to consider alternatives for evaluating the vector $\tilde{\mathbf{a}}$.

To determine the $\tilde{a}_j$, a least-square algorithm can be used to minimize the error

$$E = \int_{s[\tilde{\mathbf{u}}]} \left[\tilde{\mathbf{u}} - \sum_{j=1}^{m} \tilde{\mathbf{a}}_j \psi_j \right]^2 d\mathbf{x} \tag{19.8}$$

where the inner product is evaluated on the support $s[\tilde{\mathbf{u}}]$, thus making the two terms in the integral of the same size N. The minimizing solution to the error (19.8) requires the residual to be orthogonal to each mode ψ_n so that

$$\left(\tilde{\mathbf{u}} - \sum_{j=1}^{m} \tilde{\mathbf{a}}_j \psi_j, \psi_j \right)_{s[\tilde{\mathbf{u}}]} = 0 \qquad j \neq k,\ j = 1, 2, \cdots, m. \tag{19.9}$$

In practice then, we can project the full state vector $\mathbf{u}$ on to the support space and determine the vector $\tilde{\mathbf{a}}$:

$$\mathbf{M}\tilde{\mathbf{a}} = \mathbf{f} \tag{19.10}$$

where the matrix $\mathbf{M}$ elements are given by (19.7b) and the components of the vector f_k are given by $f_j = \left(\mathbf{u}, \psi_j\right)_{s[\tilde{\mathbf{u}}]}$. Note that in the event the measurement space is sufficiently dense, or as the support space is the entire space, then $\mathbf{M} = \mathbf{I}$, thus implying the eigenvalues of $\mathbf{M}$ approach unity as the number of measurements become dense. Once the vector $\tilde{\mathbf{a}}$ is determined, then a reconstruction of the solution can be performed us

$$\mathbf{u}(x, t) \approx \mathbf{\Psi} \tilde{\mathbf{a}}\,. \tag{19.11}$$

As the measurements become dense, the matrix $\mathbf{M}$ converge to $\mathbf{I}$ and $\tilde{\mathbf{a}} \rightarrow \mathbf{a}$.

It only remains to consider the efficacy of the measurement matrix $\mathbf{P}$. Originally, random measurements were proposed [28]. However, the ROMs community quickly developed principled techniques based upon, for example, minimization of the condition number of $\mathbf{M}$ [45], selection of minima or maxima of POD modes [46], and/or greedy algorithms of EIM/DEIM [6, 18]. Thus m measurement locations where judiciously chosen for the task of accurately interpolating the nonlinear terms in the ROM. This type of sparsity has been commonly used throughout the ROMs community. Indeed, continued efforts have been made to improve interpolation strategies, such as the recent Q-DEIM architecture [23], generalized EIM [35] or online refinement using a genetic algorithm [43].

19.4 Machine Learning and POD Libraries

The POD reduction with sparse sampling provides a number of advantages for model reduction of nonlinear dynamical systems. POD provides a principled way to construct an r-dimensional subspace $\boldsymbol{\Psi}_r$ characterizing the dynamics while sparse sampling augments the POD method by providing a method to evaluate the problematic nonlinear terms using an m-dimensional subspace projection matrix $\mathbf{P}$. Thus a small number of points can be sampled to approximate the nonlinear terms in the ROM. Figure 19.1 illustrates the library building procedure whereby a dynamical regime is sampled in order to construct an appropriate POD basis $\boldsymbol{\Psi}_r$. Inspired by machine learning methods [8, 36], the various POD basis for a parametrized system are merged into a master library of POD modes $\boldsymbol{\Psi}_L$ which contains all the low-rank subspaces exhibited by the dynamical system.

The method proposed here capitalizes on these methods by building low-dimensional libraries associated with the full nonlinear system dynamics as well as the specific nonlinearities. Interpolation points, as will be shown in what follows, can be used with sparse representation and compressive sensing to (1) identify dynamical regimes, (2) reconstruct the full state of the system, and (3) provide an efficient nonlinear model reduction and POD-Galerkin prediction for the future state.

The concept of library building of low-rank "features" from data is well established in the computer science community. In the reduced-order modeling community, it has recently become an issue of intense investigation. Indeed, a variety of recent works have produced libraries of ROM models [2, 9, 10, 19, 38–40, 42] that can be selected and/or interpolated through measurement and classification. Alternatively, cluster-based reduced order models use a k-means clustering to build a Markov transition model between dynamical states [31]. These recent innovations are similar to the ideas advocated here. However, the focus of this work is on

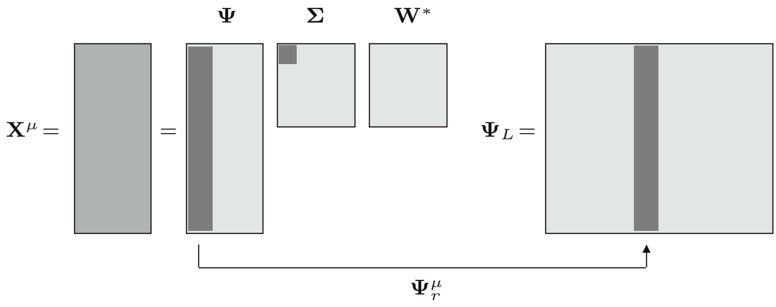


Fig. 19.1 Library construction from numerical simulations of the governing equations (19.1). Simulations are performed of the parametrized system for different values of a bifurcation parameter μ. For each regime, low-dimensional POD modes $\boldsymbol{\Psi}_r$ are computed via an SVD decomposition. The various rank-r truncated subspaces are stored in the library of modes matrix $\boldsymbol{\Psi}_L$. This is the learning stage of the algorithm

determining how suitably chosen **P** can be used across all the libraries for POD mode selection and reconstruction. If one chooses, one can build two sets of libraries: one for the full dynamics and a second for the nonlinearity so as to make it computationally efficient with the DEIM strategy [42]. Before these more formal techniques based upon machine learning were developed, it was already realized that parameter domains could be decomposed into subdomains and a local ROM/POD computed in each subdomain. Patera et al. [25] used a partitioning based on a binary tree whereas Amsallem et al. [1] used a Voronoi Tessellation of the domain. Such methods were closely related to the work of Du and Gunzburger [24] where the data snapshots were partitioned into subsets and multiple reduced bases computed. The multiple bases were then recombined into a single basis, so it doesn't lead to a library per se. For a review of these domain partitioning strategies, please see Ref. [3].

19.5 Compressive Sensing for POD Mode Selection

Although there are a number of techniques for selecting the correct POD library elements to use, including the workhorse *k*-means clustering algorithm [2, 19, 38–40], we will instead make use of sparse sampling and compressive sensing innovations for characterizing the nonlinear dynamical system [9, 10, 42]. Compressive sensing has emerged as a leading theoretical construct for using a sparse number of samples for reconstructing a full state space [5, 12–15, 21, 22]. Specifically, we wish to use a limited number of sensors for classifying the dynamical regime of the system from a range of potential POD library elements characterized by a parameter μ. Once a correct classification is a achieved, a standard ℓ_2 reconstruction of the full state space can be accomplished with the selected subset of POD modes, and a POD-Galerkin prediction can be computed for its future.

In general, we will have a sparse measurement vector $\tilde{\mathbf{u}}$ given by (19.4). The full state vector $\mathbf{u}$ can be approximated with the POD library modes ($\mathbf{u} = \boldsymbol{\Psi}_L \mathbf{a}$), therefore

$$\tilde{\mathbf{u}} = \mathbf{P}\boldsymbol{\Psi}_L \mathbf{a}, \tag{19.12}$$

where $\boldsymbol{\Psi}_L$ is the low-rank matrix whose columns are POD basis vectors concatenated across all β regimes and $\mathbf{c}$ is the coefficient vector giving the projection of $\mathbf{u}$ onto these POD modes. If $\mathbf{P}\boldsymbol{\Psi}_L$ obeys the restricted isometry property [5] and $\mathbf{u}$ is sufficiently sparse in $\boldsymbol{\Psi}_L$, then it is possible to solve the highly-underdetermined system (19.12) with the sparsest vector $\mathbf{a}$. Mathematically, this is equivalent to an ℓ_0 optimization problem which is *np*-hard. However, under certain conditions, a sparse solution of Eq. (19.12) can be found by minimizing the l_1 norm instead [14, 22] so that

$$\mathbf{c} = \arg\min_{\mathbf{a}'} ||\mathbf{a}'||_1, \quad \text{subject to} \quad \tilde{\mathbf{u}} = \mathbf{P}\boldsymbol{\Psi}_L \mathbf{a}. \tag{19.13}$$

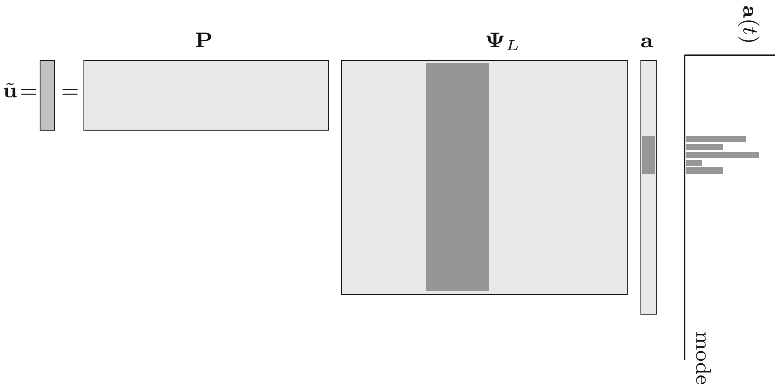


Fig. 19.2 The compressive sensing algorithm for mode selection. In this mathematical framework, a sparse measurement is taken of the system (19.1) and a highly under-determined system of equations $\mathbf{P}\mathbf{\Psi}_L\mathbf{a} = \tilde{\mathbf{u}}$ is solved subject to ℓ_1 penalization so that $\|\mathbf{a}\|_1$ is minimized. Illustrated is the selection of the μth POD modes. The bar plot on the left depicts the non-zero values of the vector $\mathbf{a}$ which correspond to the $\mathbf{\Psi}_r$ library elements. Note that the sampling matrix $\mathbf{P}$ that produces the sparse sample $\tilde{\mathbf{u}} = \mathbf{P}\mathbf{u}$ is critical for success in classification of the correct library elements $\mathbf{\Psi}_r$ and the corresponding reconstruction

The last equation can be solved through standard convex optimization methods. Thus the ℓ_1 norm is a proxy for sparsity. Note that we only use the sparsity for classification, not reconstruction. Figure 19.2 demonstrate the sparse sampling strategy and prototypical results for the sparse solution $\mathbf{a}$. For a review of compressive sensing, sparse representation and sparsity promoting methods, see recent comprehensive reviews on the subject [26, 27].

19.6 Example: Flow Around a Cylinder

To demonstrate the sparse classification and reconstruction algorithm developed, we consider the canonical problem of flow around a cylinder. This problem is well understood and has already been the subject of studies concerning sparse spatial measurements [9, 11, 32, 44]. Specifically, it is known that for low to moderate Reynolds numbers, the dynamics is spatially low-dimensional and POD approaches have been successful in quantifying the dynamics [20, 29, 33, 34, 44]. The Reynolds number, *Re*, plays the role of the bifurcation parameter μ in (19.1), i.e. it is a parametrized dynamical system.

The data we consider comes from numerical simulations of the incompressible Navier-Stokes equation:

$$\frac{\partial u}{\partial t} + u \cdot \nabla u + \nabla p - \frac{1}{Re}\nabla^2 u = 0 \tag{19.14a}$$

$$\nabla \cdot u = 0 \tag{19.14b}$$

where $u\,(x, y, t) \in \mathbb{R}^2$ represents the 2D velocity, and $p\,(x, y, t) \in \mathbb{R}^2$ the corresponding pressure field. The boundary condition are as follows: (1) Constant flow of $u = (1, 0)^T$ at $x = -15$, i.e., the entry of the channel, (2) Constant pressure of $p = 0$ at $x = 25$, i.e., the end of the channel, and (3) Neumann boundary conditions, i.e. $\frac{\partial u}{\partial \mathbf{n}} = 0$ on the boundary of the channel and the cylinder (centered at $(x, y) = (0, 0)$ and of radius unity).

For each relevant value of the parameter Re we perform an SVD on the data matrix in order to extract POD modes. It is well known that for relatively low Reynolds number, a fast decay of the singular values is observed so that only a few POD modes are needed to characterize the dynamics. Figure 19.3 shows the 3 most dominant POD modes for Reynolds number $Re = 40, 150, 300, 1000$. Note that 99% of the total energy (variance) is selected for the POD mode selection cut-off, giving a total of $1, 3, 3, 9$ POD modes to represent the dynamics in the regimes shown. For a threshold of 99.9%, more modes are required to account for the variability.

Classification of the Reynolds number is accomplished by solving the optimization problem (19.13) and obtaining the sparse coefficient vector **a**. Note that each entry in **a** corresponds to the energy of a single POD mode from our library. For simplicity, we select a number of local minima and maxima of the POD modes as sampling locations [46] for the matrix **P**. Because the optimization in ℓ_1 promotes sparsity, those coefficients from POD modes associated with the measured flow primarily are large the nonzero terms. The classification of the Reynolds number is done by summing the absolute value of the coefficient that corresponds to each Reynolds number. To account for the large number of coefficients allocated for the higher Reynolds number (which may be 16 POD modes for 99.9% variance at $Re = 1000$, rather than a single coefficient for Reynolds number 40), we divide by the square root of the number of POD modes allocated in **a** for each Reynolds number. The classified method is the one that has the largest magnitude after this process. The result of this classification process is summarized in Table 19.1.

Although the accuracy in classification is quite high, many of the false classifications are due to categorizing a Reynolds number from a neighboring flow, i.e. the Reynolds 1000 is often mistaken for Reynolds number 800. This is largely due to the fact that these two Reynolds numbers are strikingly similar so that the algorithm proposed has a difficult time separating their modal structures. Figure 19.4 shows a schematic of the sparse sensing configuration along with the reconstruction of the pressure field achieved at $Re = 1000$ with 15 sensors. Classification and reconstruction performance can be improved using other methods for constructing

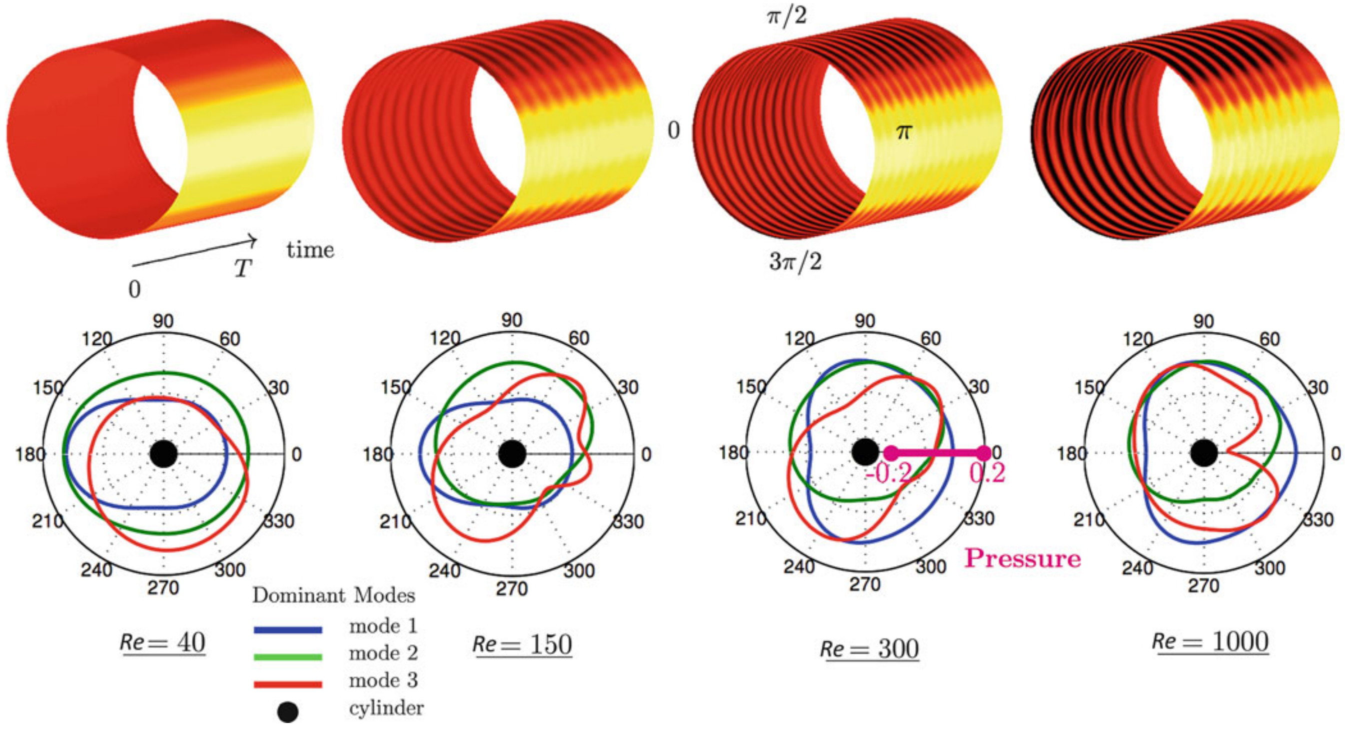


Fig. 19.3 Time dynamics of the pressure field (*top panels*) for flow around a cylinder for Reynolds number $Re = 40, 150, 300$ and 1000. Collecting snapshots of the dynamics reveals low-dimensional structures dominate the dynamics. The dominant three POD pressure modes for each Reynolds number regime are shown in polar coordinates. The pressure scale is in magenta (*bottom left*)

Table 19.1 Success Rate with m sensors using random sampling and sensors placed on randomly selected minima/maxima of POD modes [46]

	m-Random sensors (%)	m-Sensors by POD (%)
$m = 10,\ r = 99.9\%$	78.9	75.4
$m = 10,\ r = 99.99\%$	48.7	76.9
$m = 20,\ r = 99.9\%$	76.1	92.9
$m = 20,\ r = 99.99\%$	85.2	88.3

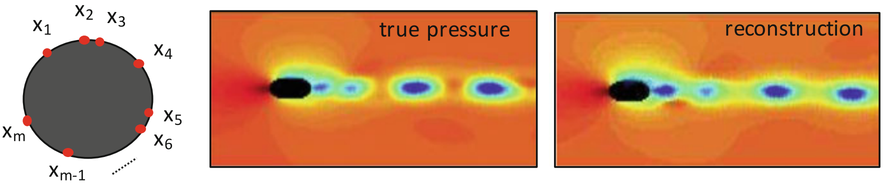


Fig. 19.4 Illustration of m sparse sensor locations (*left panel*) for classification and reconstruction of the flow field. The selection of sensory/interpolation locations can be accomplished by various algorithms [9, 11, 32, 42, 44]. For a selected algorithm, the sensing matrix **P** determines the classification and reconstruction performance

the sensing matrix **P** [9, 11, 32, 42, 44]. Regardless, this example demonstrate the usage of sparsity promoting techniques for POD mode selection (ℓ_1 optimization) and subsequent reconstruction (ℓ_2 projection).

19.7 Outlook on Sparsity for ROMs

In this work, we have shown that sparsity can be taken advantage of in at least two distinct ways: (1) for approximating the nonlinearities in the Galerkin projection through ℓ_2 interpolation, and (2) for classifying the dynamical regime of the parametrized dynamical system using ℓ_1-based compressive sensing. The former is well known in the ROMs community as gappy POD and its variants (e.g. EIM and DEIM). The latter is only now emerging as a critically enabling machine learning technique for classification and compressive architectures. In combination, the two methods are well matched as both can be leveraged to full advantage by optimizing the sampling locations of the measurement matrix **P** [42]. Indeed, poor selection of the interpolating points requires a much higher number of interpolation points m.

The full power of ℓ_1-based optimization techniques still remains an area of active research. At its core, the ℓ_1 norm serves as a proxy for sparsity, which is known to be a hallmark feature of reduced order models, i.e. r-dimensional, low-rank representations are ubiquitous. Thus in addition to classification, one might envision using sparsity promoting techniques to perform such tasks as compressive SVDs, for instance, for computing approximate POD modes using down sampled data matrices, i.e. random SVD. As computational science continues into the exascale regime, sparsity promoting techniques will only continue to grow in importance. Indeed, we envision that making full use of sparse sampling will be critically enabling for solving such high-dimensional systems. In this work, two advantageous applications are shown. Undoubtedly, more key applications will emerge for sparse sampling techniques capable of characterizing the full state space.

Acknowledgements J.N. Kutz would like to acknowledge support from the Air Force Office of Scientific Research (FA9550-15-1-0385).

References

1. Amsallem, D., Cortial, J., Farhat, C.: On demand CFD-based aeroelastic predictions using a database of reduced-order bases and models. In: AIAA Conference (2009)
2. Amsallem, D., Tezaur, R., Farhat, C.: Real-time solution of computational problems using databases of parametric linear reduced-order models with arbitrary underlying meshes. J. Comput. Phys. **326**, 373–397 (2016)
3. Amsallem, D., Zahr, M.J., Washabaugh, K.: Fast local reduced basis updates for the efficient reduction of nonlinear systems with hyper-reduction. Adv. Comput. Math. (2015). doi:10.1007/s10444-015-9409-0
4. Astrid, P.: Fast reduced order modeling technique for large scale LTV systems. In: Proceedings of 2004 American Control Conference, vol. 1, pp. 762–767 (2004)
5. Baraniuk, R.G.: Compressive sensing. IEEE Signal Process. Mag. **24**(4), 118–120 (2007)
6. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. C. R. Math. Acad. Sci. Paris **339**, 667–672 (2004)
7. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. **57**, 483–531 (2015)
8. Bishop, C.: Pattern Recognition and Machine Learning. Springer, Berlin (2006)
9. Bright, I., Lin, G., Kutz, J.N.: Compressive sensing based machine learning strategy for characterizing the flow around a cylinder with limited pressure measurements. Phys. Fluids **25**, 127102 (2013)
10. Brunton, S.L., Tu, J.H., Bright, I., Kutz, J.N.: Compressive sensing and low-rank libraries for classification of bifurcation regimes in nonlinear dynamical systems. SIAM J. Appl. Dyn. Syst. **13**, 1716–1732 (2014)
11. Brunton, B.W., Brunton, S.L., Proctor, J.L., Kutz, J.N.: Sparse sensor placement optimization for classification. SIAM J. App. Math. **76**, 2099–2122 (2016)
12. Candès, E.J.: In: Sanz-Solé, M., Soria, J., Varona, J.L., Verdera, J. (eds.) Compressive sensing. In: Proceeding of the International Congress of Mathematicians, vol. 2, pp. 1433–1452 (2006)
13. Candès, E.J., Tao, T.: Near optimal signal recovery from random projections: universal encoding strategies? IEEE Trans. Inf. Theory **52**(12), 5406–5425 (2006)
14. Candès, E.J., Romberg, J., Tao, T.: Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information. IEEE Trans. Inf. Theory **52**(2), 489–509 (2006)
15. Candès, E.J., Romberg, J., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Commun. Pure Appl. Math. **59**(8), 1207–1223 (2006)
16. Carlberg, K., Farhat, C., Cortial, J., Amsallem, D.: The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. J. Comput. Phys. **242**, 623–647 (2013)

17. Carlberg, K., Barone, M., Antil, H.: Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction. J. Comput. Phys. **330**, 693–734 (2017)
18. Chaturantabut, S., Sorensen, D.: Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. **32**, 2737–2764 (2010)
19. Choi, Y., Amsallem, D., Farhat, C.: Gradient-based constrained optimization using a database of linear reduced-order models. arXiv:1506.07849 (2015)
20. Deane, A.E., Kevrekidis, I.G., Karniadakis, G.E., Orszag, S.A.: Low-dimensional models for complex geometry flows: application to grooved channels and circular cylinders. Phys. Fluids **3**, 2337 (1991)
21. Donoho, D.L.: Compressed sensing. IEEE Trans. Inf. Theory **52**(4), 1289–1306 (2006)
22. Donoho, D.L.: For most large underdetermined systems of linear equations the minimal 1-norm solution is also the sparsest solution. Commun. Pure Appl. Math. **59**(6), 797–829 (2006)
23. Drmać, Z., Gugercin, S.: A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions. SIAM J. Sci. Comput. **38**(2), A631–A648 (2016)
24. Du, Q., Gunzburger, M.: Model reduction by proper orthogonal decomposition coupled with centroidal Voronoi tessellation. In: Proceedings of the Fluids Engineering Division Summer Meeting, FEDSM2002–31051. The American Society of Mechanical Engineers (2002)
25. Eftang, J.L., Patera, A.T., Rønquist, E.M.: An HP certified reduced-basis method for parameterized elliptic PDEs. In: SIAM SISC (2010)
26. Elad, M.: Sparse and Redundant Representations. Springer, Berlin (2010)
27. Eldar, Y., Kutyniok, G. (eds.): Compressed Sensing: Theory and Applications. Cambridge University Press, Cambridge (2012)
28. Everson, R., Sirovich, L.: Karhunen-Loéve procedure for gappy data. J. Opt. Soc. Am. A **12**, 1657–1664 (1995)
29. Galletti, B., Bruneau, C.H., Zannetti, L.: Low-order modelling of laminar flow regimes past a confined square cylinder. J. Fluid Mech. **503**, 161–170 (2004)
30. Holmes, P.J., Lumley, J.L., Berkooz, G., Rowley, C.W.: Turbulence, Coherent Structures, Dynamical Systems and Symmetry. Cambridge Monographs in Mechanics, 2nd edn. Cambridge University Press, Cambridge (2012)
31. Kaiser, E., Noack, B.R., Cordier, L., Spohn, A., Segond, M., Abel, M., Daviller, G., Osth, J., Krajnovic, S., Niven, R. K.: Cluster-based reduced-order modelling of a mixing layer. J. Fluid Mech. **754**, 365–414 (2014)
32. Kaspers, K., Mathelin, L., Abou-Kandil, H.: A machine learning approach for constrained sensor placement. American Control Conference, Chicago, IL, July 1–3, 2015 (2015)
33. Liberge, E., Hamdouni, A.: Reduced order modelling method via proper orthogonal decomposition (POD) for flow around an oscillating cylinder. J. Fluids Struct. **26**, 292—311 (2010)
34. Ma, X., Karniadakis, G.E.: A low-dimensional model for simulating three-dimensional cylinder flow. J. Fluid Mech. **458**, 181–190 (2002)
35. Maday, Y., Mula, O.: A generalized empirical interpolation method: application of reduced basis techniques to data assimilation. Anal. Numer. Partial Differ. Eqn. **4**, 221–235 (2013)
36. Murphy, K.: Machine Learning: A Probabilistic Perspective. MIT Press, Cambridge (2012)
37. Nguyen, N.C., Patera, A.T., Peraire, J.: A "best points" interpolation method for efficient approximation of parametrized functions. Int. J. Num. Methods Eng. **73**, 521–543 (2008)
38. Peherstorfer, B., Willcox, K.: Online adaptive model reduction for nonlinear systems via low-rank updates. SIAM J. Sci. Comput. **37**(4), A2123–A2150 (2015)
39. Peherstorfer, B., Willcox, K.: Dynamic data-driven reduced-order models. Comput. Methods Appl. Mech. Eng. **291**, 21–41 (2015)
40. Peherstorfer, B., Willcox, K.: Detecting and adapting to parameter changes for reduced models of dynamic data-driven application systems. Proc. Comput. Sci. **51**, 2553–2562 (2015)
41. Quarteroni, A., Rozza, G. (eds.): Reduced Order Methods for Modeling and Computational Reduction. Springer, Berlin (2014)

42. Sargsyan, S., Brunton, S.L., Kutz, J.N.: Nonlinear model reduction for dynamical systems using sparse optimal sensor locations from learned nonlinear libraries. Phys. Rev. E **92**, 033304 (2015)
43. Sargasyan, S., Brunton, S.L., Kutz, J.N.: Online interpolation point refinement for reduced order models using a genetic algorithm. arxiv:1607.07702
44. Venturi, D., Karniadakis, G.E.: Gappy data and reconstruction procedures for flow past cylinder. J. Fluid Mech. **519**, 315–336 (2004)
45. Willcox, K.: Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. Comput. Fluids **35**, 208–226 (2006)
46. Yildirim, B., Chryssostomidis, C., Karniadakis, G.E.: Efficient sensor placement for ocean measurements using low-dimensional concepts. Ocean Model. **273**(3–4), 160–173 (2009)

Chapter 20
A HJB-POD Approach to the Control of the Level Set Equation

Alessandro Alla, Giulia Fabrini, and Maurizio Falcone

Abstract We consider an optimal control problem where the dynamics is given by the propagation of a one-dimensional graph controlled by its normal speed. A target corresponding to the final configuration of the front is given and we want to minimize the cost to reach the target. We want to solve this optimal control problem via the dynamic programming approach but it is well known that these methods suffer from the "curse of dimensionality" so that we can not apply the method to the semi-discrete version of the dynamical system. However, this is made possible by a reduced-order model for the level set equation which is based on Proper Orthogonal Decomposition. This results in a new low-dimensional dynamical system which is sufficient to track the dynamics. By the numerical solution of the Hamilton-Jacobi-Bellman equation related to the POD approximation we can compute the feedback law and the corresponding optimal trajectory for the nonlinear front propagation problem. We discuss some numerical issues of this approach and present a couple of numerical examples.

A. Alla (✉)
Department of Scientific Computing, Florida State University, 400 Dirac Science Library, Tallahassee, FL 32306, USA
e-mail: aalla@fsu.edu

G. Fabrini
Department of Mathematics and Statistics, University of Konstanz, Universitätsstraße 10, 78457 Konstanz, Germany
e-mail: giulia.fabrini@uni-konstanz.de

M. Falcone
Dipartimento di Matematica, Università di Roma "La Sapienza", P. A. Moro, 5, 00185 Roma, Italy
e-mail: falcone@mat.uniroma1.it

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_20

20.1 Introduction

The optimal control of partial differential equations is a challenging problem that has a rather long tradition nowadays. The origin dates back to the monograph by J.L. Lions [17] and several books on infinite dimensional optimal control problems have appeared since then (see e.g. [11, 15, 16, 21]).

Here we follow a different approach based on Dynamic Programming (DP). Although formally one can follow this approach and obtain the characterization of the value function as the unique solution (in a weak sense) of a Hamilton-Jacobi equation in a Banach space several difficulties arise when one tries to apply this approach for numerical purposes. It is well known that the DP approach suffers of the curse of dimensionality so the naive discretization based on a finite difference/finite elements discretization of the dynamics in order to reduce it to a finite dimensional problem and apply DP to this system is simply unfeasible. The number of dimensions of the discrete system will be too high (thousands or millions of nodes). This motivates a reduced order modeling approach where the dynamics will be (hopefully) represented by a low number of basis elements, then we can apply DP to this low order system. The same approach has been introduced and studied starting from the seminal paper [13, 14] and has shown to be rather effective for the optimal control of parabolic and advection-diffusion equations [2]. In the above cases the solution of the dynamics is typically regular and the asymptotic behavior is easy to predict. More recently a technique based on spectral elements has been applied to optimal control problems for the wave equation (see [12]).

The novelty in this paper is to deal with the dynamics given by the level set equation for front propagation problems, a problem with many applications in combustion, gas dynamics, fluid dynamics and image processing. The front propagation problem has solutions which are just Lipschitz continuous since singularities and topology changes in the front can appear during the evolution. Its solution must be understood in the viscosity sense (see [19] and [18] for an extensive presentation of the level set method and its applications). This clearly introduces some technical difficulties and makes it more complicated to construct the model reduction approximation based on the snapshots. We will use a model reduction based on POD (Proper Orthogonal Decomposition, [22]) to obtain a rather accurate approximation for the level-set dynamics in dimension 1. Moreover, we mention that in the paper [10] the level set method is coupled to a Reduced Basis model in order to derive a rigorous approximation of the admissible region for a system characterized by several parameters.

To set the paper into perspective, we want to mention that the problem of solving the controlled level-set equation in dimension 1 has been studied in [8], where they apply iterative descent methods for the optimization. Starting from the results obtained in [7] for the uncontrolled dynamics, they prove the existence of optimal controls under different assumptions on the speed function (which in their case is a function of the space). Concerning the solution of the control problem they give a proof in a specific setting (see [8] for all the details). The difference here is that the control is a general function of space and time and not necessarily piecewise constant (as in [8]), moreover in this paper we apply the DP approach in order to obtain an approximate feedback control. The drawback is that since we are in a general setting both for the control and the profile we want to reach, there is not a theoretical result which ensures that the controllability problem has a solution.

The paper is organized as follows: in Sect. 20.2 we present the front propagation problem with the associated optimal control problem, in Sect. 20.3 we give the main features of the DP approach and we explain how to deal with the model order reduction of the level set equation, finally in Sect. 20.4 we will present some numerical tests which show the efficiency of the proposed method.

20.2 A Front Propagation Problem with Target

Let us introduce our problem, we refer to [18] for more details on the topic. The dynamics will describe the front propagation of an interface via the level-set equation in $\mathbb{R}^n$. The typical situation is the following: an initial position for the front $\varGamma(0) = \varGamma_0$ (i.e. an initial surface in $\mathbb{R}^n$) is given and the front evolves driven by a force always directed in the normal direction to every point of the front. The velocity in the normal direction will be denoted by $V_\varGamma$ and the scalar speed $a(x,t)$ must keep the same sign during the evolution (let us choose the positive sign to fix ideas). Note that in the general case the speed can also depend on the position x and the time t, although also the case of a piecewise constant speed is interesting (and we will use it in the sequel). To summarize, we will have in general

$$V_\varGamma = a(x,t), \qquad a : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}_+. \tag{20.1}$$

The initial configuration of the front is

$$\varGamma(0) = \varGamma_0 \tag{20.2}$$

and $\varGamma_0$ can be a single closed curve or the union of arbitrarily many finite closed curves without intersections. The evolutive equation should describe the propagation of the front $\varGamma(t)$ in time. This problem can produce singularities for a single smooth curve $\varGamma_0$ even in the particular case of a constant speed. It is well known that a powerful method to track this evolution even when one has singularities and topology changes (e.g. when the front $\varGamma(t)$ starting from $\varGamma_0$ can produce intersections at time t) is the level set method where one describes $\varGamma(t)$ as the 0-level set of a continuous function solving the Cauchy problem

$$\begin{cases} \varPhi_t(x,t) + a(x,t)|\nabla\varPhi(x,t)| = 0 & \text{in } \mathbb{R}^n \times \mathbb{R}_+ \\ \varPhi(x,0) = \varPhi_0(x) & \text{in } \mathbb{R}^n \end{cases} \tag{20.3}$$

where Φ_0 is a representation function for Γ_0 (i.e. a continuous function changing sign across Γ_0) and $a(x,t)$ is assumed to be known. Solving (20.3) one can obtain $\Phi(x,t)$ and recover the position of the front $\Gamma(t)$ as

$$\Gamma(t) := \{x \in \mathbb{R}^n : \Phi(x,t) = 0\} \tag{20.4}$$

The Cauchy problem has a unique viscosity solution under rather general assumptions (see [6]).

Since here we just want to describe our technique we will consider the evolution of a graph, i.e. the dimension will be $n = 1$ and we look for the front

$$\Gamma(t) := \{(x, y(x,t)) | x \in \mathbb{R}\} \subset \mathbb{R}^2.$$

In this particular case the dynamics will be given by

$$\begin{cases} y_t = a(x,t)\sqrt{1+y_x^2}, & (x,t) \in \mathbb{R} \times [0,T], \\ y(x,0) = y_0(x), & x \in \mathbb{R}. \end{cases} \tag{20.5}$$

Many numerical schemes have been proposed so far to solve the level set Eq. (20.5). In particular we refer to [19] for monotone and consistent schemes based on finite difference approximation and to [9] for semi-Lagrangian schemes. In the present work, we will adopt an explicit finite difference scheme. We closely follow the scheme used in [7]; we choose space and time steps, Δx and Δt respectively and let

$$x_j = j\Delta x,\ j \in \mathbb{Z}, \quad t_n = n\Delta t,\ 0 \le n \le N,$$

where $N\Delta t = T$. We denote by y_j^n the numerical approximation of $y(x_j, t_n)$. We approximate the solution of Eq. (20.5) using the following scheme: let $y_j^0 = y_0(x_j)$, $j \in \mathbb{Z}$ and for $n = 0, \dots, N-1$

$$y_j^{n+1} = y_j^n + \Delta t\, a(x_j, t_n)\sqrt{1 + \max\left\{\left(\frac{y_{j-1}^n - y_j^n}{\Delta x}, \frac{y_{j+1}^n - y_j^n}{\Delta x}\right)\right\}^2} \quad j \in \mathbb{Z}.$$

More information about the numerical approximation can be found [8]. Let us remark that we must work on a bounded interval $\Omega := (a,b)$ for numerical purposes. Then the grid will have only a finite number of nodes $a = x_0 < x_1 < \ldots < x_d = b$ and, in order to give enough freedom to the front evolution, we impose homogeneous zero Neumann boundary conditions (see [19] for more details on the implementation).

Let us introduce the control problem for the front propagation. Now the speed function $a(x,t)$ will not be fixed but will be our control function which we can vary in order to steer the solution as close as possible to a particular desired configuration of the front, e.g. our *target* denoted by $\bar{y}$. In this framework, the speed $a(x,t)$ will be denoted as $u(x,t)$, adopting the classical notation for control problems. In conclusion we have a control problem for a first order nonlinear partial differential equation of Hamilton-Jacobi type which can develop singularities during the evolution. This is known to be a difficult problem for the lack of regularity of the solution. Note that another important issue is the reachability of the target: we are not aware of any theoretical result which ensure us that the target is reachable in finite time so it is natural to set the problem as an infinite horizon problem. We will use the corresponding cost functional with a quadratic running cost in order to penalize the distance from the target:

$$J_p(y_0, u(t)) = \int_0^\infty \|y(x,t) - \bar{y}\|_p^2 \chi_{\bar{y}}(x) e^{-\lambda t} dt, \tag{20.6}$$

where $y(x,t)$ is the solution of (20.5), ε is a positive parameter and

$$\chi_{\bar{y}}(x) = \begin{cases} 1 & \mathit{if}\ \|y(x,t) - \bar{y}\|_p > \varepsilon \\ 0 & \mathit{otherwise}. \end{cases} \tag{20.7}$$

Note that there is a strong dependence of the cost function from the initial condition $y_0(x)$ and from the norm of the running cost p. In fact we want to solve an infinite horizon optimal control problem with a running cost which penalizes the distance in L^p-norm (where $p = 1, 2, \infty$) from our target which is a stripe of radius ε centered in the profile we want to reach $\bar{y}$. For a given time $t > 0$ and $\Omega = [a, b] \subset \mathbb{R}$ we define the L^∞-error as

$$||y(x,t) - \bar{y}||_\infty := \max_{x \in \Omega} |y(x,t) - \bar{y}|$$

and the L^p-error ($p = 1, 2$) as

$$||y(x,t) - \bar{y}||_p := \left(\int_\Omega |y(x,t) - \bar{y}|^p dx \right)^{\frac{1}{p}}.$$

Let us also observe that the characteristic function (20.7) makes the costs vanish whenever we enter a neighborhood of the target. The reachability of the target is an interesting open problem which we will not address in this paper (however in the numerical examples the neighborhood is always reachable).

20.3 An HJB-POD Method for the Control of the Level-Set Equation

In this section we recall the main features of the dynamic programming principle and Proper Orthogonal Decomposition (POD). Finally, we also explain the coupling of the two methods. The interested reader can find more details in [3, 14].

20.3.1 Numerical Approximation of the HJB Equation

We illustrate the dynamic programming approach for optimal control problems of the form

$$\min_{u \in \mathcal{U}} J(y_0, u) := \int_0^\infty L(y(s), u(s))\, e^{-\lambda s}\, ds \tag{20.8}$$

constrained by the nonlinear ordinary differential equation:

$$\begin{cases} \dot{y}(t) = f(y(t), u(t)), & t > 0, \\ y(0) = y_0 \end{cases} \tag{20.9}$$

with system dynamics in $\mathbb{R}^n$ and a control signal $u(t) \in \mathcal{U} \equiv \{u(\cdot)\text{ measurable}, u : [0, +\infty[\to U\}$, where U is a compact subset of $\mathbb{R}^m$; we assume $\lambda > 0$, while $L(\cdot, \cdot)$ and $f(\cdot, \cdot)$ are Lipschitz-continuous, bounded functions. More details on this topic can be found in [5]. In this setting, a standard tool is the application of the dynamic programming principle, which leads to a characterization of the value function

$$v(y_0) := \inf_{u \in \mathcal{U}} J(y_0, u) \tag{20.10}$$

as unique viscosity solution of the HJB equation:

$$\lambda v(y_0) + \sup_{u \in U}\{-\nabla v \cdot f(y_0, u) - L(y_0, u)\} = 0, \tag{20.11}$$

where Dv is the gradient of the value functions. Equation (20.11) may be approximated in several ways, we consider a fully-discrete semi-Lagrangian scheme which is based on the discretization of the system dynamics with time step h, and a mesh parameter k, leading to a fully discrete approximation $V_{h,k}(y_0)$ satisfying

$$V_{h,k}(y_{0_i}) = \min_{u \in U}\{(1 - \lambda h) I_1[V_{h,k}](y_{0_i} + hf(y_{0_i}, u)) + L(y_{0_i}, u)\}\,, \tag{20.12}$$

for every element y_{0_i} of the discretized state space. Note that in general, the arrival point $y_{0_i} + hf(y_{0_i}, u)$ is not a node of the state space grid, and therefore the value is computed by means of a linear interpolation operator, denoted by $I_1[V_{h,k}]$.

The bottleneck of this approach is related to the so-called *curse of the dimensionality*, namely, the computational cost increases dramatically as soon as the dimension does. One way to overcome the dimensionality issue is the construction of efficient iterative solvers for (20.12).

The simplest iterative solver is based on a direct fixed point iteration of the value function known as value iteration method (VI)

$$V_{h,k}^{j+1} := \min_{u \in U}\{(1-\lambda h)I_1[V_{h,k}^j](y_{0_i} + hf(y_{0_i}, u)) + L(y_{0_i}, u)\}\,, \quad i = 1, \ldots, N_p \tag{20.13}$$

where N_p denotes the number of nodes of the grid. This algorithm converges slowly for any initial condition $V_{h,k}^0$ since the operator on the right hand side is a contraction mapping.

A more efficient formulation is the so-called *policy iteration algorithm* (PI), which starting from an initial guess u_i^0 of the control at every node, performs the following iterative procedure:

$$\begin{aligned} [V_{h,k}^j]_i &= (1-\lambda_h)I_1[V_{h,k}^j](y_{0_i} + hf(y_{0_i}, u_i^j)) + hL(y_{0_i}, u_i^j) \\ [u^{j+1}]_i &= \operatorname*{argmin}_{u \in U}\{(1-\lambda h)I_1[V_{h,k}^j](y_{0_i} + hf(y_{0_i}, u)) + hL(y_{0_i}, u)\} \end{aligned}$$

where we first have to solve a linear system, since we freeze the control, in order to find the value function corresponding to the given control and then update the control. We iterate until convergence to the value function. The PI algorithm has a quadratic convergence provided a good initial guess is given and its convergence is only local (as for the Newton method), so there is a need for good initialization. In order to provide a smart initial guess for the algorithm it was proposed in [2] an acceleration mechanism based on a (VI) solution on a coarse grid, which is used to generate an initial guess for (PI) on the fine grid. The proposed coupling aims at efficiency and robustness. In this work, we adopt the Accelerated Policy Iteration method (shortly API) for the approximation of the HJB equation (see [2] for more details).

The main advantage of the dynamic programming approach is the possibility to have a synthesis of feedback controls. Once the discretized value function $V_{h,k}$ has been obtained, the approximated optimal control $u_{h,k}^*(y_0)$ for a point y_0 of the state space is obtained by:

$$u_{h,k}^*(y_0) = \arg\min_{u \in U}\{(1-\lambda h)I_1[V_{h,k}](y_0 + hf(y_0, u)) + L(y_0, u)\} \tag{20.14}$$

This choice is quasi-optimal provided some additional condition on the dynamics are satisfied, a typical example is a linear dependence on the control variable as it has been shown in [9, p. 231]. Finally, let us observe that our optimal control problem fits into the general framework if we define in (20.8) and (20.9), respectively:

$$L(y(t), u(t)) := \|y(t) - \bar{y}\|_p^2 \chi_{\bar{y}},$$

$$f(y_j^n, u) := u(x_j, t)\sqrt{1 + \max\left\{\left(\frac{y_{j-1}^n - y_j^n}{\Delta x}, \frac{y_{j+1}^n - y_j^n}{\Delta x}\right)\right\}^2}.$$

20.3.2 POD Approximation of the Control Problem

In this section, we explain the POD method for the approximate solution of the optimal control problem. The approach is based on projecting the nonlinear dynamics onto a low dimensional manifold utilizing projectors which contain information of the dynamics. A common approach in this framework is based on the snapshot form of POD proposed in [20], which works as follows.

The snapshots are computed by the numerical approximation of (20.9) for $y(t_i) \in \mathbb{R}^n$ for given time instances and a reference control. Its choice turns out to be very important in order to build accurate surrogate model and may provide basis function which are not able to capture the desired dynamics.

We define the POD ansatz of order ℓ for the state y as

$$y(t) \approx \bar{y} + \sum_{i=1}^{\ell} y_i^\ell(t)\psi_i \,. \tag{20.15}$$

where $\bar{y} \in \mathbb{R}^n$ is our target. We define the snapshot matrix $Y = [y(t_0)-\bar{y}, \ldots, y(t_n)-\bar{y}]$ and determine its singular value decomposition $Y = W\Sigma V$. The POD basis functions $\Psi = \{\psi_i\}_{i=1}^\ell$ of rank ℓ are the first ℓ columns of the matrix W.

The reduced optimal control problem is obtained through replacing (20.9) by a dynamical system obtained from a Galerkin approximation with basis functions $\{\psi_i\}_{i=1}^\ell$ and ansatz (20.15) for the state.

This leads to an ℓ-dimensional system for the unknown coefficients $\{y_i^\ell\}_{i=1}^\ell$, namely

$$\dot{y}^\ell(t) = \Psi^T f(\Psi y^\ell, u(t)), \quad y^\ell(0) = y_0^\ell. \tag{20.16}$$

where $y_0^\ell = \Psi^T(y_0 - \bar{y}) \in \mathbb{R}^\ell$. The error of the Galerkin projection is governed by the singular values associated to the truncated states of the SVD.

The POD-Galerkin approximation leads to the optimization problem

$$\inf_{u\in\mathscr{U}} J_{y_0^\ell}^\ell(u) := \int_0^\infty L(y^\ell(s), u(s))e^{-\lambda s}\, ds, \tag{20.17}$$

where $u \in \mathscr{U}$, y^ℓ solves the reduced dynamics (20.16). The value function v^ℓ, defined for the initial state $y_0^\ell \in \mathbb{R}^\ell$ is given by

$$v^\ell(y_0^\ell) = \inf_{u\in\mathscr{U}} J_{y_0^\ell}^\ell(u)\,.$$

Note that the resulting HJB equations are defined in $\mathbb{R}^\ell$, but for computational purposes we need to restrict our numerical domain to a bounded subset of $\mathbb{R}^\ell$. We refer the interested reader to [1] for details on this issue.

20.4 Numerical Tests

In this section we describe our numerical tests. The aim is to drive an initial front's profile to a desired final configuration which will be our target, no final time is given. We compute the snapshots with an initial guess for the control inputs. We remark that it is rather crucial to obtain snapshots which simulate the desired trajectory. In fact, to the best of our knowledge, there is no general recipe even for linear dynamics and this is an open question which is hard to address in general. In the current work, we could observe the sensitivity of the surrogate model with respect to the choice of the initial input. However, we found very helpful to enrich the snapshot set with the desired configuration $\bar{y}$. A study of basis generation in this framework may be found in [4]. To apply model order reduction we assume that the control may be rewritten as follows:

$$u(x,t) := \sum_{i=1}^{M} u_i(t) b_i(x) \tag{20.18}$$

where $u_i : [0, +\infty] \to U$ are the control inputs, M is the finite number of control functions which will be used to reconstruct $u(x,t)$ and the coefficients $b_i : \mathbb{R}^n \to \mathbb{R}$ are the so called shape functions which model the actions that we can applied to the system governed by our model. The dynamics is given by (20.5) and we performed the simulations choosing different norms in the cost functional in (20.6).

To show the effectiveness of the method we compute the error in different norms between the final configuration of the front and the given target. We define the error as follows:

$$\mathscr{E}_p = \|y_f(x) - \bar{y}\|_p, \qquad p = 1, 2, \infty \tag{20.19}$$

where we denote $y_f(x)$ the final configuration of the front. All the numerical simulations have been realized on a MacBook Pro with 1 CPU Intel Core i5 2.4 GHz and 8GB RAM. The codes used for the simulations are written in Matlab.

20.4.1 Test 1: Constant Final Configuration

In this test we choose the initial profile $y_0(x) = 1 + \frac{cos(2\pi(1-x))}{2}$ in (20.5) with $x \in [0,1]$. We want to steer the front toward the target $\bar{y}(x) = 2.5$. We compute the snapshots with a finite difference explicit scheme with a space step $\Delta x = 0.05$, time step $\Delta t = 0.01$ and a given input $u(x,t) = 0.42e^{-(x-0.5)^2}$. The shape functions in (20.18) are $b_1(x) = y_0(x), b_2(x) = e^{-(x-0.5)^2}$ and the control set is $U = [-2, 2]$. In this test the chosen parameters for the value function are: $\Delta x = 0.1, \varepsilon = 0.01, \lambda = 1, \ell = 5, \Delta\tau = 0.01$ (the time step to integrate

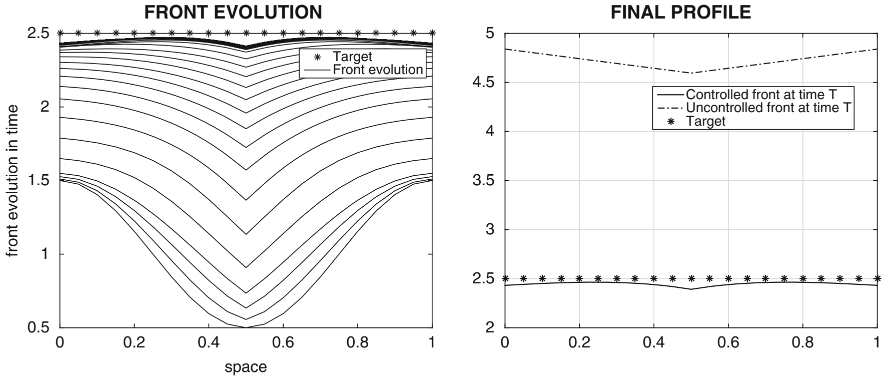


Fig. 20.1 Test 1: evolution of the controlled front in the phase-plane with the target (*left*), final controlled and uncontrolled front's profile with the target (*right*) with $p = 2$

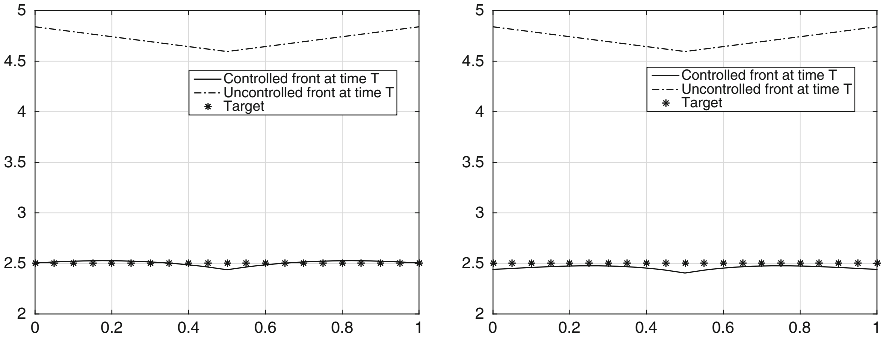


Fig. 20.2 Test 1: Final controlled and uncontrolled front's profile and target using the norm $p = 1$ (*left*) and the norm $p = \infty$ (*right*)

the trajectories). The set U is discretized into 9 equidistant elements for the value function and 21 for the trajectories.

In the left panel of Fig. 20.1 we show the controlled evolution of the front. We can observe that the final configuration of the front is in a neighborhood of the desired configuration. In the right panel of Fig. 20.1 we compare the controlled front's configuration, obtained with the L^2-norm with the target and the uncontrolled front. Figure 20.2 shows the same comparison where the optimal configuration is computed with L^1 and L^∞-norm. Although the classical choice for the norm in the cost functional is $p = 2$, we obtain better results for $p = 1$. We also consider $p = \infty$.

In Table 20.1 we compute the quantity $\mathscr{E}_p$ to evaluate the distance between the controlled final configuration and the desired one in different norms. We also evaluate the cost functional with different choices of p. It turns out that the norm with $p = 1$ provides the most accurate final configuration, whereas the norm $p = 2$ has lower value of the cost functional. We note that the evaluation of the cost functional takes into account the whole history of the trajectories and not just the final configuration.

Table 20.1 Test 1: Error between final and desired configuration and evaluation of the cost functional for $\varepsilon = 0.01$

	$p = 1$	$p = 2$	$p = \infty$
$\mathscr{E}_p$	0.0214	0.0584	0.0949
$J_p(y_0, u)$	0.3326	0.3185	0.5832

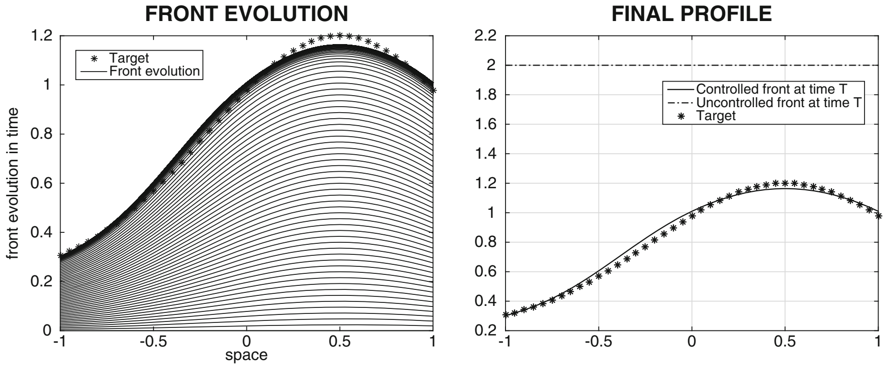


Fig. 20.3 Test 2: evolution of the front in the phase-plane with the target (*left*), final controlled and uncontrolled front's profile with the target (*right*) with $p = 2$

20.4.2 Test 2: Constant Initial Configuration

In this test we choose a constant initial profile $y_0(x) \equiv 0$ in (20.5) with $x \in [-1, 1]$. The target is $\bar{y}(x) = 0.2 + e^{-(x-0.5).^2}$. We compute the snapshots with a finite difference explicit scheme with a space step $\Delta x = 0.05$, time step $\Delta t = 0.01$ and velocity $u(x,t) = 0.2 + e^{-(x-0.5).^2}$.

In this test the parameters for the value function are: $\Delta x = 0.1, \varepsilon = 0.01, \lambda = 1, U \equiv [0, 2], b(x) = 0.2 + e^{-(x-0.5).^2}, \ell = 4, \Delta\tau = 0.01$ (the time step to integrate the trajectories). The number of controls are 11 for the value function and 21 for the trajectories.

In Fig. 20.3 we show the evolution of the controlled front where the final profile is steered close to the target.

For the sake of completeness we also show the optimal control in Fig. 20.4. As explained in Test 1, we perform the simulations using different norms in the cost functional ($p = 1, 2, \infty$). Table 20.2 shows the distance between the controlled solution and the desired configuration and the evaluation of the cost functional. Here, we can see that the choice of $p = 2$ in the norm for the cost functional provides the most accurate final configuration, whereas $p = \infty$ provides a lower value for the cost functional.

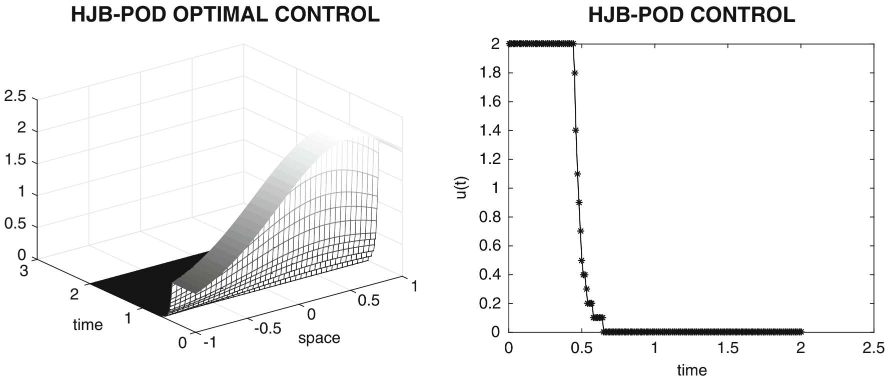


Fig. 20.4 Test 2: evolution of HJB-POD control in time $u(t)$ (*left*), evolution of HJB-POD control $u(x, t)$ (*right*)

Table 20.2 Test 2: Error between final and desired configuration and evaluation of the cost functional for $\varepsilon = 0.01$

	$p = 1$	$p = 2$	$p = \infty$
$\mathscr{E}_p$	0.0526	0.0439	0.0617
$J_p(y_0, u)$	0.2561	0.2562	0.2218

20.4.3 Test 3: A Non-regular Target

Here we consider a final configuration which is not regular. To this end let us define

$$\bar{y}(x) := C_1\chi_{[a,\bar{x}]}(x) + C_2\chi_{[\bar{x},b]}(x). \qquad (20.20)$$

The constant initial profile is $y_0(x) \equiv 0$ in (20.5) with $x \in [0, 1]$. We compute the snapshots with a finite difference explicit scheme with a space step $\Delta x = 0.05$, time step $\Delta t = 0.01$ and velocity $u(x, t) = C_1\chi_{[0,\bar{x}]} + C_2\chi_{[\bar{x},1]}$, with $C_1 = 0.5, C_2 = 0.8.\bar{x} = 0.5$.

In this test the parameters for the value function are: $\Delta x = 0.1, \varepsilon = 0.01, \lambda = 1$, $U \equiv [0, 3], b_1(x) = \chi_{[0,\bar{x}]}, b_2 = \chi_{[\bar{x},1]}$ (shape functions), $\ell = 4$ (POD basis's rank) $\Delta\tau = 0.01$ (the time step to integrate the trajectories).

The number of controls are 16 for the value function and 31 for the trajectories. In Fig. 20.5 we show the evolution of the controlled front where the final profile is steered close to the target with $p = 2$. Finally, in Fig. 20.6 we also show the optimal control.

An analysis of the distance between the controlled and desired configuration is provided in Table 20.3. In this example, we can see that the norm with $p = 2$ provides the most accurate solution for the final configuration and the cost functional. Then, the results with $p = 1, +\infty$ are displayed in Fig. 20.7.

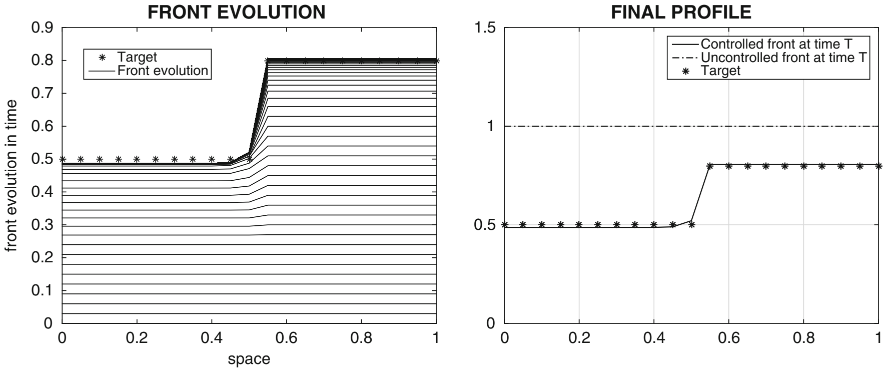


Fig. 20.5 Test 3: evolution of the front in the phase-plane with the target (*left*), final controlled and uncontrolled front's profile with the target (*right*) with $p = 2$

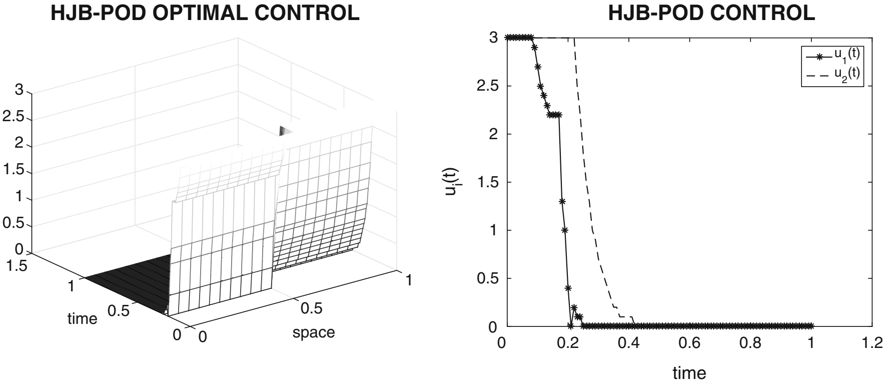


Fig. 20.6 Test 3: evolution of HJB-POD control in time $u(t)$ (*left*), evolution of HJB-POD control $u(x, t)$ (*right*)

Table 20.3 Test 3: Error between final and desired configuration and evaluation of the cost functional with $\varepsilon = 0.01$

	$p = 1$	$p = 2$	$p = \infty$
$\mathcal{E}_p$	0.0256	0.011	0.0218
$J_p(y_0, u)$	0.0382	0.0366	0.0568

20.5 Conclusions

We have proposed a HJB-POD approach for the control of a nonlinear hyperbolic problem which typically has weak solutions in the viscosity sense. This problem is more difficult with respect to other evolutive problems, such as parabolic equations, where the regularity of the initial condition is preserved or even improved. Therefore, it is not trivial that POD model order reduction with a few number of basis

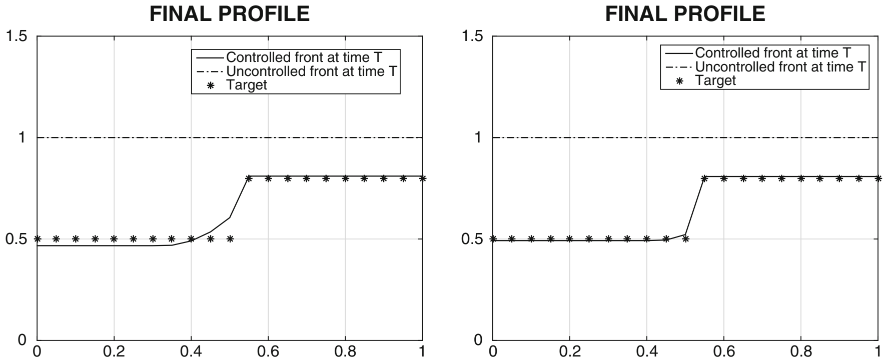


Fig. 20.7 Test 3: final controlled and uncontrolled front's profile and target using the norm $p = 1$ (*left*) and the norm $p = \infty$ (*right*)

functions provide a satisfactory approximation of the model. Indeed, numerical simulations show that if we represent the front with a POD-basis with rank 4 or 5 we obtain satisfactory results. Clearly it would be helpful to increase the number of basis functions to obtain better performances by the HJB-POD method (see [3]). Un fortunately, that is very hard at the moment due to the curse of dimensionality of dynamic programming, another possibility is to up-date the set of basis functions during the evolution as in [1]. Furthermore, we have investigated different norms in the cost functional, motivated by the lack of general theory particularly for nonlinear dynamics. It turns out that the best approximation is obtained using the standard L^2 norm in most of the cases.

The computation of the basis functions remains an open question that definitely deserves further investigation. We will try to extend the results in [3] to build theoretical results in a future work.

Acknowledgements The first author is supported by US Department of Energy grant number DE-SC0009324.

References

1. Alla, A., Falcone, M.: An adaptive POD approximation method for the control of advection-diffusion equations. In: Kunisch, K., Bredies, K., Clason, C., von Winckel, G. (eds.) Control and Optimization with PDE Constraints. International Series of Numerical Mathematics, vol. 164, pp. 1–17. Birkhäuser, Basel (2013)
2. Alla, A., Falcone, M., Kalise, D.: An efficient policy iteration algorithm for dynamic programming equations. SIAM J. Sci. Comput. **37**, 181–200 (2015)
3. Alla, A., Falcone, M., Volkwein, S.: Error Analysis for POD approximations of infinite horizon problems via the dynamic programming principle. SIAM J. Control. Optim. (submitted, 2015)

4. Alla, A., Schmidt, A., Haasdonk, B.: Model order reduction approaches for infinite horizon optimal control problems via the HJB equation. In: Benner, P., et al. (eds.) Model Reduction of Parametrized Systems. MS&A, vol. 17. Springer International Publishing, Cham (2017). doi:10.1007/978-3-319-58786-8_21
5. Bardi, M., Capuzzo Dolcetta, I.: Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations. Birkhauser, Basel (1997)
6. Barles, G.: Solutions de Visocité des Equations de Hamilton-Jacobi. Springer, Berlin (1994)
7. Deckelnick, K., Elliott, C.M.: Propagation of graphs in two-dimensional inhomogeneous media. Appl. Numer. Math. **56**, 3, 1163–1178 (2006)
8. Deckelnick, K., Elliott, C.M., Styles, V.: Optimal control of the propagation of a graph in inhomogeneous media. SIAM J. Control. Optim. **48**, 1335–1352 (2009)
9. Falcone, M., Ferretti, R.: Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi Equations. SIAM, Philadelphia (2014)
10. Grepl, M., Veroy, K.: A level set reduced basis approach to parameter estimation. C. R. Math. **349**, 1229–1232 (2011)
11. Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications, vol. 23. Springer, Berlin, (2009)
12. Kröner, A., Kunisch, K., Zidani, H.: Optimal feedback control of undamped wave equations by solving a HJB equation. ESAIM: Control Optim. Calc. Var. **21**, 442–464 (2014)
13. Kunisch, K., Xie, L.: POD-based feedback control of Burgers equation by solving the evolutionary HJB equation. Comput. Math. Appl. **49**, 1113–1126 (2005)
14. Kunisch, K., Volkwein, S., Xie. L.: HJB-POD based feedback design for the optimal control of evolution problems. SIAM J. Appl. Dyn. Syst. **4**, 701–722 (2004)
15. Lasiecka, I., Triggiani, R.: Control Theory for Partial Differential Equations: Continuous and Approximation Theories. In: Abstract Parabolic Systems. Encyclopedia of Mathematics and Its Applications 74, vol. I. Cambridge University Press, Cambridge (2000)
16. Lasiecka, I., Triggiani, R.: Control Theory for Partial Differential Equations: Continuous and Approximation Theories. In: Abstract Hyperbolic-Like Systems Over a Finite Time Horizon. Encyclopedia of Mathematics and Its Applications 74, vol. II. Cambridge University Press, Cambridge (2000)
17. Lions, J.L.: Optimal Control of Systems Governed by Partial Differential Equations, Band 170. Springer, New York/Berlin (1971)
18. Osher, S., Fedkiw, R.P.: Level Set Methods and Dynamic Implicit Surfaces. Springer, New York (2003)
19. Sethian, J.A.: Level Set Methods and Fast Marching Methods. Cambridge University Press, Cambridge (1999)
20. Sirovich, L.: Turbulence and the dynamics of coherent structures. Parts I-II. Q. Appl. Math. **XVL**, 561–590 (1987)
21. Tröltzsch, F.: Optimal Control of Partial Differential Equations: Theory, Methods and Application. American Mathematical Society, Providence (2010)
22. Volkwein, S.: Model reduction using proper orthogonal decomposition. Lecture Notes, University of Konstanz (2013). http://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/scripts.php

Chapter 21
Model Order Reduction Approaches for Infinite Horizon Optimal Control Problems via the HJB Equation

Alessandro Alla, Andreas Schmidt, and Bernard Haasdonk

Abstract We investigate feedback control for infinite horizon optimal control problems for partial differential equations. The method is based on the coupling between Hamilton-Jacobi-Bellman (HJB) equations and model reduction techniques. It is well-known that HJB equations suffer the so called *curse of dimensionality* and, therefore, a reduction of the dimension of the system is mandatory. In this report we focus on the infinite horizon optimal control problem with quadratic cost functionals. We compare several model reduction methods such as Proper Orthogonal Decomposition, Balanced Truncation and a new algebraic Riccati equation based approach. Finally, we present numerical examples and discuss several features of the different methods analyzing advantages and disadvantages of the reduction methods.

21.1 Introduction

The approximation of optimal control problems for partial differential equations (PDEs) is a very challenging topic. Although it has been successfully studied for open-loop problems (we address the interested reader to the books [15, 23] for more details), the closed-loop control problem presents several open questions for infinite dimensional equations.

One common way to obtain a feedback control is by means of the dynamic programming principle (DPP). The DPP characterizes the value function and its continuous version leads to a HJB equation. The theory of the viscosity solution allows us to characterize the value function as the unique solution of the HJB

A. Alla
Department of Scientific Computing, Florida State University, Tallahassee, FL, USA
e-mail: aalla@fsu.edu

A. Schmidt (✉) • B. Haasdonk
Institute for Applied Analysis and Numerical Simulation, University of Stuttgart, Stuttgart, Germany
e-mail: schmidta@mathematik.uni-stuttgart.de; haasdonk@mathematik.uni-stuttgart.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_21

equations. We note that these results are quite general and valid for any problem dimension. We refer to the book [6] for more details about the topic for ordinary differential equations. For the sake of completeness, we also mention Model Predictive Control as alternative to obtain a feedback control (see [14]).

The numerical approximation of HJB equations has been studied with different techniques such as Finite Difference, Finite Volume and Semi-Lagrangian schemes. We refer the interested reader to [12] for a comprehensive analysis of these methods.

The DPP is known to suffer the so called *curse of dimensionality*, namely the computational complexity of the problem increases exponentially when the dimension does. The problem is much harder when dealing with PDEs since their spatial discretization leads to huge systems of ODEs. Typically, we are able to solve a HJB equations numerically up to dimension 4 or 5. For this reason, model reduction plays a crucial role in order to reduce the complexity of the problem and to make the control problems feasible. The procedure is thus split in two parts, where the first part consists of finding a reduced order model (ROM) which is suitable for the control purpose, followed by the numerical solution of the HJB equations, associated with the control problem, where the full system is replaced with the ROM.

Proper Orthogonal Decomposition (POD, see [24]) and Balanced Truncation (BT, see [5]) are two of the most popular techniques for model reduction of dynamical systems, including spatially discretized PDEs. POD is a rather general method, which is based on a Galerkin projection method for nonlinear dynamical systems where the basis functions are built upon information on the system whereas the BT method is based on a Petrov-Galerkin projection, where the basis functions are obtained by solving two Lyapunov equations. The latter approach is only valid for linear systems, although extensions can be formulated (see [19]).

The coupling between HJB equations and POD has already been proposed by a series of pioneering work [17, 18]. A study of the feedback control and an adaptive method can be found in [1] and [4]. Error estimation for the method has been recently studied in [3]. We refer to [16] for the coupling with BT. For the sake of completeness, we also mention a different approach for the control of infinite dimensional systems, based on the DPP and a sparse grid approach, see [13].

In addition to POD and BT, in this work we consider a new approach based on solutions of algebraic Riccati equations (ARE) for the approximation of the value function for linear quadratic problems. This approach turns out to better capture information of the control problem and improve the quality of the suboptimal control. We analyze and compare the reduction techniques for linear and nonlinear dynamical systems. We note that in the nonlinear settings we linearize the dynamical system in a neighborhood of the desired state to apply BT and the MOR approach based on the solutions of the ARE equation.

The paper is organized as follows. In Sect. 21.2 we recall the main results on dynamic programming. Section 21.3 explains the model order reduction approaches and their application to the dynamic programming principle and the HJB equations. Finally, numerical tests are presented in Sect. 21.4 and conclusions are drawn in Sect. 21.5.

21.2 Numerical Approximation of HJB Equations

In this section we recall the basic results for the approximation of the Bellman equation, more details can be found in [6] and [12].

Let the dynamics be given by

$$\begin{cases} \dot{y}(t) &= f(y(t), u(t)), \quad t \geq 0, \\ y(0) = x, \end{cases} \tag{21.1}$$

where the state $y(t) \in \mathbb{R}^n$, the control $u(t) \in \mathbb{R}^m$ and $u \in \mathbb{U} \equiv \{u : [0, +\infty) \to U, \text{ measurable}\}$ where U is a closed bounded subset of $\mathbb{R}^m$, and $x \in \mathbb{R}^n$ is the initial condition. If f is Lipschitz continuous with respect to the state variable and continuous with respect to (y, u), the classical assumptions for the existence and uniqueness result for the Cauchy problem (21.1) are satisfied (see [6]).

The cost functional $J : \mathbb{U} \to \mathbb{R}$ we want to minimize is given by:

$$J_x(u(\cdot)) := \int_0^\infty g(y(s), u(s))\mathrm{e}^{-\lambda s} \mathrm{d}s\,, \tag{21.2}$$

where g is Lipschitz continuous in both arguments and $\lambda \geq 0$ is a given parameter. The function g represents the running costs and λ is the discount factor which guarantees that the integral is finite whenever g is bounded and $\lambda > 0$. Let us define the value function of the problem as

$$v(x) := \inf_{u(\cdot) \in \mathbb{U}} J_x(u(\cdot))\,. \tag{21.3}$$

The Dynamic Programming Principle (DPP) characterizes the value function as follows

$$v(x) = \inf_{u \in \mathbb{U}} \{ \int_0^T g(y_x(t, u), u(t))\mathrm{e}^{-\lambda t}\,\mathrm{d}t + v(y_x(T, u))\mathrm{e}^{-\lambda T}\}, \tag{21.4}$$

where $y_x(t, u)$ is the solution of the dynamics for a given initial condition x and any $T > 0$. From the DPP, one can obtain a characterization of the value function in terms of the following first order nonlinear Bellman equation

$$\lambda v(x) + \max_{u \in U}\{-f(x, u) \cdot Dv(x) - g(x, u)\} = 0, \quad \text{for } x \in \mathbb{R}^n\,. \tag{21.5}$$

Here, $Dv(x)$ denotes the gradient of v at the point x. Once the value function is computed we are able to build the feedback as follows:

$$u^*(x) := \arg\min_{u \in U}\{f(x, u) \cdot Dv(x) + g(x, u)\}.$$

Several approximation schemes on a fixed grid G have been proposed for (21.5). Here we will use a semi-Lagrangian approximation based on the Dynamic Programming Principle. This leads to

$$v_{\Delta t}(x) = \min_{u \in U}\{\mathrm{e}^{-\lambda \Delta t} v_{\Delta t}\,(x + \Delta t f\,(x,u)) + \Delta t g\,(x,u)\}\,, \tag{21.6}$$

where $v_{\Delta t}(x)$ converges to $v(x)$ when $\Delta t \to 0$. A natural way to solve (21.6) is to write it in fixed point iteration form

$$V_i^{k+1} = \min_{u \in U}\{\mathrm{e}^{-\lambda \Delta t} \mathscr{I}[V^k]\,(x_i + \Delta t f\,(x_i,u)) + \Delta t g\,(x_i,u)\}\,, \quad i = 1,\ldots,N_G. \tag{21.7}$$

Here V_i^k represents the values of the value function v at a node x_i of the grid at the k-th iteration in (21.7) and $\mathscr{I}$ is a multilinear interpolation operator acting on the values of the equidistant grid G with mesh spacing denoted by Δx.

The method is referred to in the literature as the *value iteration method*. The convergence of the value iteration can be very slow and accelerated techniques, such as the *policy iteration* technique, can be found in [2].

Remark 1 Let us mention that in general it is hard to find an explicit solution for Eq. (21.5) due to the nonlinearity of the problem. A particular case is the so called linear quadratic regulator (LQR) problem where the dynamics is linear and the cost functional is quadratic. The equations are thus given as

$$f(y,u) = Ay + Bu, \quad g(y,u) = y^T Q y + u^T R u,$$

where $A, Q \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times m}$, $R \in \mathbb{R}^{m \times m}$ with Q and R symmetric and Q positive semi-definite and R positive definite. Furthermore, the set of admissible control values is $U = \mathbb{R}$. Under these assumptions, it is known that the value function at any point $x \in \mathbb{R}^n$ is given by $v(x) = x^T P x$ where $P \in \mathbb{R}^{n \times n}$ is the solution of the following shifted algebraic Riccati equation (ARE):

$$(A - \lambda I_n)^T P + P(A - \lambda I_n) - PBR^{-1}B^T P + Q = 0. \tag{21.8}$$

Here, $I_n \in \mathbb{R}^{n \times n}$ is the n-dimensional identity matrix. Finally, the optimal control is directly given in an appropriate state-feedback form $u(t) = -R^{-1}B^T P y(t)$. More details on the LQR can be found in [10]. We will use the LQR problem for comparison purposes as a benchmark model in the numerical examples, see Sect. 21.4.

21.3 Model Reduction

The focus of this section is to recall some model reduction techniques utilized to build surrogate models in this work. The Reduced Order Modelling (ROM) approach to optimal control problems is based on projecting the nonlinear dynamics onto a low dimensional manifold utilizing projectors that contain information of the expected controlled dynamics. The idea behind the projection techniques is to restrict the dynamics $y(t)$ onto a low-dimensional subspace $\mathscr{V} \subset \mathbb{R}^n$ that contains the relevant information. We equip the space $\mathscr{V}$ with a basis matrix $V \in \mathbb{R}^{n\times\ell}$ which will be specified in the following subsections, and approximate the full state vector by $y(t) \approx Vy^\ell(t)$, where $y^\ell(t) : [0,\infty) \to \mathbb{R}^\ell$ are the reduced coordinates. Plugging this ansatz into the dynamical system (21.1), and requiring a so called Petrov-Galerkin condition yields

$$\begin{cases} \dot{y}^\ell(t) = W^T f(Vy^\ell(t), u(t)) \\ y^\ell(0) = W^T x, \end{cases} \tag{21.9}$$

where the matrix $W \in \mathbb{R}^{n\times\ell}$ is chosen, such that $W^T V = I_\ell$. Further sampling based techniques can be employed to obtain an efficient scheme for nonlinear problems as suggested in [9, 11] and the references therein. The presented procedure is a generic framework for model reduction. It is clear, that the quality of the approximation greatly depends on the reduced space $\mathscr{V}$. In the next subsections, we briefly revisit some classical projection techniques and introduce a new approach, which is tailored for the approximation of the value function.

21.3.1 Proper Orthogonal Decomposition

A common approach is based on the snapshot form of POD proposed in [21], which in the present situation works as follows. We compute a set of snapshots $y_1, \dots, y_k$ of the dynamical system (21.1) corresponding to a prescribed input and different time instances $t_1, \dots, t_k$ and define the POD ansatz of order ℓ for the state $y(t)$ by

$$y(t) \approx \sum_{i=1}^{\ell} y_i^\ell(t)\psi_i, \tag{21.10}$$

where the basis vectors $\{\psi_i\}_{i=1}^\ell$ are obtained from the singular value decomposition of the snapshot matrix $Y = [y_1, \dots, y_k]$, i.e. $Y = \Psi\Sigma\Gamma$, and the first ℓ columns of $\Psi = (\Psi_1, \dots, \Psi_n)$ form the POD basis functions of rank ℓ. Hence we choose the basis vectors $V = W = (\Psi_1, \dots, \Psi_\ell)$ for the POD method in (21.9). Here the SVD is based on the Euclidean inner product. This is reasonable in our situation, since the numerical computations performed in our examples are based on finite difference schemes.

In the present work the quality of the resulting basis is strongly related to the choice of a given input u, whose optimal choice is usually unknown. For control problems, one way to improve this selection is to compute snapshots from the following equation for a given pair (y, u) and any final time $T > 0$

$$-\dot{p}(t) = f_y(y(t), u(t))p(t) + g_y(y(t), u(t)), \quad p(T) = 0, \tag{21.11}$$

as suggested in [22]. We refer to $p : [0, T] \to \mathbb{R}^n$ as the adjoint solution (see [15]). The advantage of this approach is that it is able to capture the dynamics of the adjoint Eq. (21.11) which is directly related to the optimality conditions. In order to obtain the POD basis, one has to simulate the high dimensional system and subsequently perform a SVD, which can both be implemented very efficiently.

21.3.2 Balanced Truncation

The balanced truncation (BT) method is a well-established ROM technique for LTI systems

$$\begin{aligned} \dot{y}(t) &= Ay(t) + Bu(t), \\ z(t) &= Cy(t), \end{aligned}$$

where $z(t)$ is the output of interest. We refer to [5] for a complete description of the topic. The BT method is based on the solution of the reachability Gramian $\tilde{P}$ and the observability Gramian $\tilde{Q}$ which solve respectively the following Lyapunov equations

$$A\tilde{P} + \tilde{P}A^T + BB^T = 0, \quad A^T\tilde{Q} + \tilde{Q}A + C^TC = 0.$$

We determine the Cholesky factorization of the Gramians

$$\tilde{P} = \Phi\Phi^T, \qquad \tilde{Q} = \Upsilon\Upsilon^T.$$

Then, we compute the singular value decomposition of the Hankel operator $\Upsilon^T\Phi$ and set

$$W = \Upsilon U_1 \Sigma_1^{1/2}, \qquad V = \Upsilon V_1 \Sigma_1^{1/2},$$

where $U_1, V_1 \in \mathbb{R}^{n\times\ell}$ are the first ℓ columns of the left and right singular vectors of the Hankel operator and $\Sigma_1 = \mathrm{diag}(\sigma_1, \dots, \sigma_\ell)$ matrix of the first ℓ singular values.

The idea of BT is to neglect states that are both, hard to reach and hard to observe. This is done by neglecting states that correspond to low Hankel singular values σ_i. This method is very popular, also because the whole procedure can be verified by a-priori error bounds in several system norms, and the Lyapunov equations can be solved very efficiently due to their typical low-rank structure in large-scale applications, see [7].

21.3.3 A New Approach Based on Algebraic Riccati Equations

For arbitrary control problems, the value function is in general not available in analytical form. However, in the case of the LQR problem, the value function has the quadratic form $v(x) = x^T Px$ where P solves an algebraic Riccati equation (21.8).

Thus, the relevant information of the value function is stored in the matrix P and can be extracted by taking the SVD (or eigenvalue decomposition, since P is symmetric) $P = \Psi \Sigma \Psi^T$ with an orthonormal matrix $\Psi = [\psi_1, \dots, \psi_n]$. We can approximate P with $P^\ell = \sum_{k=1}^{\ell} \sigma_k \psi_k \psi_k^T$ and the error bound reads

$$\|P - P^\ell\|_2 \le \sigma_{\ell+1},$$

where we applied the Schmidt-Eckart-Young-Mirsky theorem as mentioned in [5, 20]. We define the reduced value function as $v^\ell(x) := x^T P^\ell x$. Then the following bound holds true

$$|v(x) - v^\ell(x)| \le \sigma_{\ell+1} \|x\|^2, \quad \forall x \in \mathbb{R}^n.$$

Thus, if we define the reduced space $\mathscr{V} := \operatorname{span}(\psi_1, \dots, \psi_\ell)$, we can expect an accurate approximation of the relevant information in the value function, at least in the case where the system dynamics are linear. Finally, we note that in (21.9) we choose $V = W = (\Psi_i)_{i=1}^{\ell}$. This procedure thus requires the solution of an algebraic Riccati equation and a subsequent SVD. By employing low-rank techniques, both of these tasks can be sped up substantially.

21.3.4 The Coupling Between HJB and Model Reduction

Since the curse of dimensionality prohibits a direct solution of the HJB equations in higher dimensions, we apply model reduction in the first place, in order to obtain a small system for which the HJB equation admits a computable solution. We note that model reduction does not solve the curse of dimensionality. However, we choose the number of basis functions in such a way that the reduced problem becomes computationally tractable, unlike the full-dimensional problem which cannot be solved numerically. In the general projection framework above, we define the following reduced HJB problem, which is the optimal control problem for the projected system:

$$\inf_{u \in \mathscr{U}} J^\ell_{W^T x}(u) = \inf_{u \in \mathscr{U}} \int_0^\infty g(Vy^\ell(t), u(t), t) \mathrm{e}^{-\lambda t} \, \mathrm{d}t, \tag{21.12}$$

$$\text{s.t.} \quad \begin{aligned} \dot{y}^\ell(t) &= W^T f(Vy^\ell(t), u(t)), \quad t \ge 0 \\ y^\ell(0) &= W^T x \end{aligned} \tag{21.13}$$

As in the full-dimensional case, the value function $v^\ell(W^T x) = \inf_{u\in\mathscr{U}} J^\ell_{W^T x}(u)$ fulfills an ℓ-dimensional HJB equation, which can be solved numerically. This gives an approximation to the true (in general unknown) value function at the point $x \in \mathbb{R}^n$:

$$\hat{v}^\ell(x) := v^\ell(W^T x). \tag{21.14}$$

Furthermore, the reduced value function $\hat{v}^\ell(x)$ can be used to define a reduced feedback control function similar to the full dimensional case as

$$\hat{u}^*(x) := \min_{u\in U}\{f(x,u)\cdot D\hat{v}^\ell(x) + g(x,u)\}.$$

Remark 2 For the numerical approximation of the value function, we must restrict our computational domain in the ℓ-dimensional reduced space. Since the physical meaning of the full-coordinates is lost when going to the reduced coordinates, it is in general not clear how to choose the interval lengths of the grid. We therefore restrict ourselves to the approximation of the value function for vectors in the set $\Theta := \{x \in \mathbb{R}^n \text{ s.t. } \|x\|_\infty \le a\}$, i.e. for all $x \in \Theta$ and $i = 1, \ldots, n$ it holds $|x_i| \le a$, where x_i denotes the i-th component of x. We then define the reduced domain

$$\Theta_\ell := \bigtimes_{i=1}^{\ell} (\underline{\mathrm{x}}_i, \bar{x}_i) \subset \mathbb{R}^\ell, \tag{21.15}$$

where the interval boundaries $\underline{\mathrm{x}}_i$ and $\bar{x}_i$ are calculated in such a way that for all full states $x \in \Theta$, the projected vectors are mapped to vectors in Θ_ℓ, i.e. $W^T x \in \Theta_\ell$ for all $x \in \Theta$. Thus, we expect to have a valid value function for all vectors $x \in \Theta$. A different approach for the reduced interval can be found in [1].

21.4 Numerical Examples

We now compare the different approaches introduced in Sect. 21.3. The first example is a classical LQR scenario, i.e. a linear system with quadratic cost functional. This simple setup has the huge advantage of a known value function, that can be used for comparing the different approaches for the HJB approximations. In the second example, we study the behavior of the feedback control for a nonlinear viscous Burgers equation.

21.4.1 One-Dimensional Heat Advection-Diffusion Equation

Our first example consists of a one-dimensional advection-diffusion equation

$$\begin{aligned}
\partial_t w(t,\xi) - \mu_{\text{diff}}\partial_{\xi\xi} w(t,\xi) + \mu_{\text{adv}}\partial_{\xi} w(t,\xi) &= \mathbf{1}_{\Omega_B}(\xi)u(t), && t \geq 0, \xi \in \Omega\\
w(t,\xi) &= 0, && t \geq 0, \xi \in \partial\Omega\\
w(0,x) &= w_0(x), && x \in \Omega\\
z(t) &= \frac{1}{|\Omega_C|}\int_{\Omega_C} w(t,\xi)\mathrm{d}\xi,
\end{aligned}$$

with $\Omega := (-1,1) \subset \mathbb{R}$, $\partial\Omega = \{-1,1\}$ and distributed control acting on a set $\Omega_B = [-0.5,-0.1]$. The output of interest $z(t)$ is the average temperature distribution on the interval $\Omega_C = [0.1, 0.6]$, $\mathbf{1}_{\Omega_B}(\xi)$ and $\mathbf{1}_{\Omega_C}(\xi)$ denote the characteristic functions of the set Ω_B resp. Ω_C at the point $\xi \in \Omega$. We choose the parameter values $\mu_{\text{diff}} = 0.2$ and $\mu_{\text{adv}} = 2$. We discretize the PDE in space by using a finite difference scheme on an equidistant grid with interior points $\{\xi_i\}_{i=1}^n$. The dimension of the semi-discrete problem is 61. The advection term is discretized by using an upwind scheme. In order to solve the problem numerically for the simulation and the generation of the snapshots, we apply an explicit Euler scheme with $\Delta t = \frac{1}{2}\mu_{\text{diff}}\Delta x^2$. In order to obtain a control problem, we introduce the cost functional as in Remark 1 with $Q = 20C^TC$ and $R = 0.1$, where C is the discretized representation of $z(\cdot)$. The final setting is given by

$$\min_{u\in L^2(0,\infty)} \int_0^\infty (20z(t)^2 + 0.1u(t)^2)\mathrm{d}t$$

$$\text{s.t.} \quad \dot{y}(t) = f(y(t),u(t)) = Ay(t) + Bu(t), \quad z(t) = Cy(t), \quad y(0) = x.$$

The solution to this problem can be calculated in a closed loop form and is given by $u(t) = -10B^TPx(t)$, where $P \in \mathbb{R}^{n\times n}$ solves the associated ARE (21.8) with $\lambda = 0$. Furthermore, the value function is known to be a quadratic function of the form $v(x) = x^TPx$. Figure 21.1 shows the controlled and uncontrolled solution for the initial condition $x = \left(0.2 \cdot \mathbf{1}_{(-0.8,-0.6)}(\xi_i)\right)_{i=1}^n$, where the true LQR control is used to generate the figure.

We now construct the bases W_q and V_q for the different approaches $q \in \{$POD, PODadj, BT, Ricc$\}$ introduced in Sect. 21.3. In order to obtain the basis for the POD approach, we simulate the full system with a prescribed control function $u(t) = \sin(t)$ for $t \in [0, 2\pi]$ and compute the POD method as explained in Sect. 21.3. Since W_{POD} is an orthonormal matrix, we simply set the biorthogonal counterpart as $V_{\text{POD}} := W_{\text{POD}}$. The basis $W_{\text{POD,adj}}$ for the adjoint system is calculated with the same control input and discretization parameters, but solving Eq. (21.11). The basis matrices for balanced truncation are denoted as W_{BT} and V_{BT} and are calculated in the usual way as explained in Sect. 21.3. Finally, the Riccati basis is

Fig. 21.1 Initial state, uncontrolled and controlled state of the linear example at time $t = 0.3$

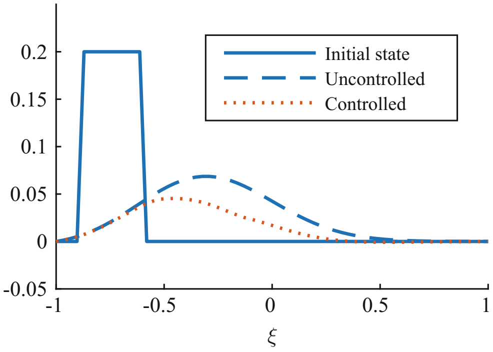
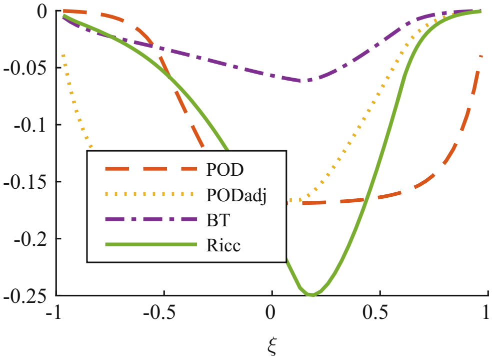


Fig. 21.2 Dominant basis vectors for all approaches



built by taking the first ℓ left singular vectors of the SVD of $P \in \mathbb{R}^{n \times n}$, where P solves the ARE (21.8).

We now calculate the reduced value functions for the different approaches, which we will denote as $\hat{v}_q^\ell$ with q as above. For that purpose, we discretize the reduced domain (21.15) by dividing each dimension in 29 equidistant intervals. We then apply a value iteration scheme to calculate the solution of the HJB equations. For details, we refer to Sect. 21.2 and the references given there. The goal in this linear example is to reproduce the true LQR control and value function by the HJB approach. The set of admissible controls is thus chosen as a discrete grid on the interval $[-2, 2]$ with 301 grid points. This set of controls is sufficiently large, to capture the control values for all possible vectors $x \in \Theta$ with $a = 0.2$, see Remark 2. The fine resolution in the control space allows a good comparison of the HJB control to the true LQR control.

As a first qualitative comparison, we plot the dominant basis vectors of all different approaches in Fig. 21.2. It can be seen that the basis vectors carry very different information. Especially the basis vector for the Riccati approach does not reflect the input region of the model very well, but it provides details about the region of measurement Ω_C. Still, by its construction we expect accurate approximations of the value function.

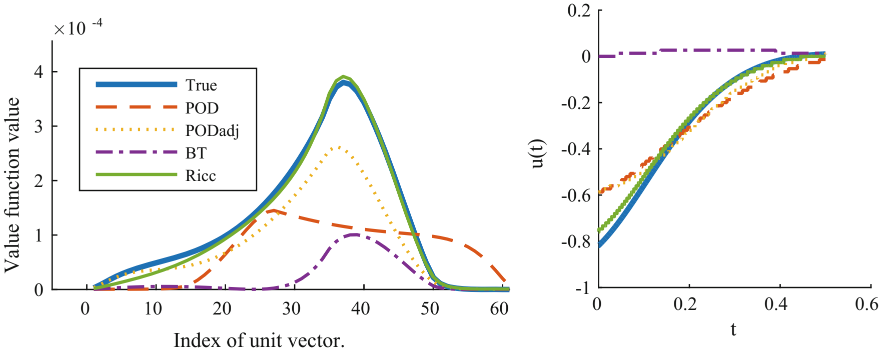


Fig. 21.3 Results for the approximation of the value function (*left*) for $\ell = 3$. True (LQR) control and approximated controls (*right*) for a given initial state $x = 0.2(B + C^T)$

Table 21.1 Mean approximation error of the value function for the different approaches

	$\ell = 1$	$\ell = 2$	$\ell = 3$	$\ell = 4$
POD	0.6426	0.5634	0.3297	0.3752
PODadj	0.8144	0.4008	0.1036	0.0959
BT	0.9971	0.8271	0.7387	0.5848
Ricc	0.5472	0.1363	0.0711	0.0566

Another interesting insight is given, when we compare the values of the approximated value functions $\hat{v}_{\ell,q}(\cdot)$ at the points $x_i := 0.2e_i$, where e_i is the i-th unit vector in $\mathbb{R}^n$. The results are depicted in Fig. 21.3 for $\ell = 3$. We see that the different bases deliver different results: the Riccati and adjoint approach capture the original behavior of the value function. We note that if we increase the dimension of the surrogate, the results improve for all approaches. In Fig. 21.3 we also show the resulting optimal control, and again we can see how the Riccati and adjoint approach are able to recover the true control signal.

A more quantitative comparison is given in Table 21.1: We calculate the values of the true value function and the reduced value functions for all approaches for 50 random test vectors from the set Θ. We next calculate the relative error between the approximation and the true LQR value function and list the mean approximation error in Table 21.1. In this example, the POD-basis does not yield accurate approximations to the true value function. Balanced truncation requires even more basis functions to capture the relevant information for the value function. This can be seen for instance by comparing the approximation error for $\ell = 4$ in the BT case to the error for $\ell = 2$ in the POD case: The BT approach requires more basis vectors to reach the same mean error. Only the adjoint approach and the basis W_{Ricc} yield very accurate results.

21.4.2 Viscous Burgers Equation

Let us now study a more complex dynamical system, where no analytical value function can be derived. We choose the 1D viscous Burgers equation on the domain $\Omega := (-1, 1)$ with homogeneous Dirichlet boundary conditions. The continuous equations now read as follows:

$$\begin{aligned} \partial_t w(\xi, t) - 0.2\partial_{\xi\xi} w(\xi, t) + 5w\partial_\xi w(\xi, t) &= \mathbf{1}_{\Omega_B}(\xi)u(\xi), \quad \xi \in \Omega, t \geq 0 \\ w(\xi, t) &= 0, \quad \xi \in \{-1, 1\}, t \geq 0, \\ w(\xi, 0) &= w_0(\xi), \quad \xi \in \Omega. \end{aligned}$$

The output of interest in this case is defined as the integral of the state over the whole domain: $z(t) := \int_\Omega w(\xi, t)\mathrm{d}\xi$ for $t \geq 0$. The control acts on the subdomain $\Omega_B := [-0.7, -0.5]$. The semi-discretization is again performed by using finite differences with the same setting as in the linear example. The discretized system has now dimension $n = 61$ and all computations are again performed by using an explicit Euler scheme. The discretized PDE and the discretized output then have the form (21.1) with

$$f(y(t), u(t)) = Ay(t) + Bu(t) + \tilde{f}(y(t)), \quad y(0) = x, \quad z(t) = Cy(t),$$

where $\tilde{f}(y)$ models the discretized nonlinear transport term.

We introduce an infinite-horizon optimal control problem, similar to the LQR case, by defining the cost functional for the discretized equations as

$$J_x(u(\cdot)) := \int_0^\infty (100z(t)^2 + 0.1u(t)^2)\mathrm{e}^{-\lambda t}\mathrm{d}t$$

with the discount factor $\lambda = 1$. Figure 21.4 shows the uncontrolled state and controlled solution. We note that the stabilization of the Burgers equation via LQR problem has been studied in [8]. The control in the latter case has been computed after a linearization of the dynamics around the set point $y = 0$ in order to solve the ARE equation. The continuous initial condition is $w_0(\xi) = 0.2(1 - \xi^2)$. The corresponding output and control function is depicted in Fig. 21.5. We can observe that the Riccati based approach is able to recover the LQR control. We recall that in the HJB setting the control space is discretized and it is not continuous as in the LQR setting.

We build the different bases for this example with the same setting as in the linear example before, only the time-steps for the HJB scheme have been adjusted and the controls are chosen as 41 equidistant points from $[-5, 5]$ in order to allow the necessary higher control values. Note that we do not aim at a direct comparison to the LQR control, we do not need to discretize the control set as fine as in the first example. For the BT and the Riccati approach, we linearize the system around

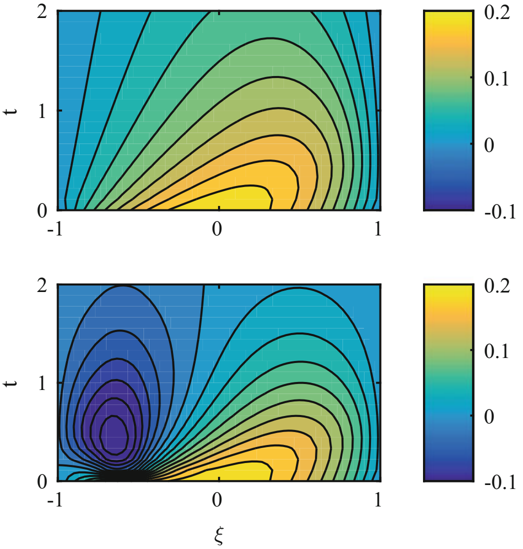


Fig. 21.4 Uncontrolled (*top*) and LQR-controlled (*bottom*) state example of the Burgers equation

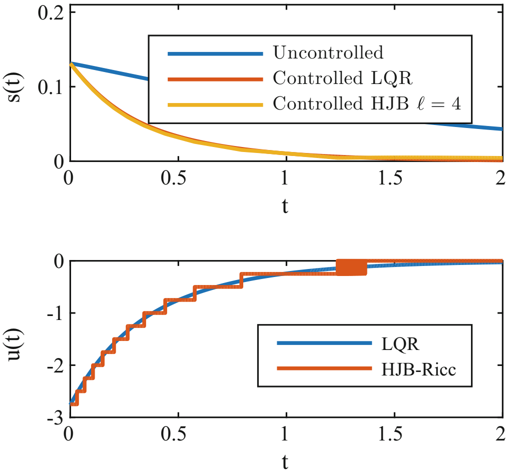


Fig. 21.5 Output of interest (*top*) and control (*bottom*) for the nonlinear Burgers example with the LQR and HJB-Riccati control for $\ell = 4$

$y = 0$ and obtain a heat equation for which the BT basis and the ARE solution are calculated. Then, the calculation of the value function is performed for the nonlinear reduced equation, where the discretization is performed in the same manner as in the linear example.

In this example we do not have a closed-loop form of the value function and thus we need a different way to compare the results. For this purpose, we approximate the value of the cost functional numerically by performing a highly-resolved simulation, followed by a quadrature using the trapezoidal rule. We simulate the closed-loop systems until $T = 5$, which suffices to neglect the increment in the cost functional on $t \in (5, \infty)$.

Table 21.2 Cost functional values for different initial vectors

	$x_{0,1}$				$x_{0,2}$			
	$\ell = 1$	$\ell = 2$	$\ell = 3$	$\ell = 4$	$\ell = 1$	$\ell = 2$	$\ell = 3$	$\ell = 4$
Ricc	0.2962	0.2958	0.2955	0.2956	0.3789	0.3786	0.3786	0.3785
POD	0.3926	0.3171	0.3112	0.3006	0.4197	0.3817	0.3802	0.3790
BT	0.2981	0.3169	0.3297	0.3260	0.3785	0.3987	0.4115	0.4080
PODadj	0.2960	0.2958	0.2955	0.2953	0.3786	0.3786	0.3786	0.3786
LQR	0.2959	0.2959	0.2959	0.2959	0.3786	0.3786	0.3786	0.3786

To compare the methods we show in Table 21.2 the evaluation of the cost functional for different initial conditions $x_{0,1} = 0.2B$ and $x_{0,2} = 0.2(1-\xi)^2$ and model reduction methods. It is hard to compare the method since we do not know the full solution, however it turns out that the Riccati and POD adjoint approach have the minimum values and are the closest to the full dimensional Riccati linearized control.

21.5 Conclusion

In this paper we propose a comparison of different model order reduction techniques for dynamic programming equations. Numerical experiments show that the POD adjoint and the Riccati based approach provide very accurate approximation for the control problem with quadratic cost functional. This is what one can expect since both methods contain information about the optimization problem, unlike BT and POD when the snapshots are generated with a random initial input. Moreover, the Riccati based approach can be generalized to nonlinear dynamics. Here we propose to linearize the system around one point of interests. In the future we would like to investigate a greedy strategy to select more points. A parametric scenario will also be considered in a future work as proposed in [20] for linear dynamical systems.

Acknowledgements The first author is supported by US Department of Energy grant number DE-SC0009324. The second and third authors thank the Baden Württemberg Stiftung gGmbH and the German Research Foundation (DFG) for financial support within the Cluster of Excellence in Simulation Technology (EXC 310/1) at the University of Stuttgart.

References

1. Alla, A., Falcone, M.: An adaptive POD approximation method for the control of advection-diffusion equations. In: K. Kunisch, K. Bredies, C. Clason, G. von Winckel (eds.) Control and Optimization with PDE Constraints. International Series of Numerical Mathematics, vol. 164, pp. 1–17. Birkhäuser, Basel (2013)
2. Alla, A., Falcone, M., Kalise, D.: An efficient policy iteration algorithm for dynamic programming equations,. SIAM J. Sci. Comput. **37**, 181–200 (2015)

3. Alla, A., Falcone, M., Volkwein, S.: Error Analysis for POD approximations of infinite horizon problems via the dynamic programming principle. SIAM J. Control Optim. (to appear)
4. Alla, A., Falcone, M., Kalise, D.: A HJB-POD feedback synthesis approach for wave equation. Bull. Braz. Math. Soc. New Ser. **47**, 51–64 (2016)
5. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. Society for Industrial and Applied Mathematics, Philadelphia, PA (2005)
6. Bardi, M., Capuzzo-Dolcetta, I.: Optimal Control and Viscosity Solutions of Hamilton-Jacobi-Bellman Equations. Birkhäuser, Basel (1997)
7. Benner, P., Saak, J.: Numerical solution of large and sparse continuous time algebraic matrix Riccati and Lyapunov equations: a state of the art survey. GAMM-Mitteilungen **36**, 32–52 (2013)
8. Burns, J., Kang, S.: A control problem for Burgers' equation with bounded input/output. Nonlinear Dyn. **2**, 235–262 (1991)
9. Chaturantabut, S., Sorensen, D.C.: Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. **32**, 2737–2764 (2010)
10. Curtain, R.F., Zwart, H.J.: An Introduction to Infinite-Dimensional Linear Systems Theory. Springer, New York (1995)
11. Drohmann, M., Haasdonk, B., Ohlberger, M.: Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. SIAM J. Sci. Comput. **34**, 937–969 (2012)
12. Falcone, M., Ferretti, R.: Semi-Lagrangian Approximation Schemes for Linear and Hamilton-Jacobi equations. Society for Industrial and Applied Mathematics, Philadelphia (2014)
13. Garcke, J., Kröner, A.: Suboptimal feedback control of PDEs by solving HJB equations on adaptive sparse grids. J. Sci. Comput. **70**(1), 1–28 (2017)
14. Grüne, L., Panneck, J.: Nonlinear Model Predictive Control: Theory and Applications. Springer, New York (2011)
15. Hinze, M., Pinnau, R., Ulbrich, M., Ulbrich, S.: Optimization with PDE Constraints. Mathematical Modelling: Theory and Applications, vol. 23. Springer, Cham (2009)
16. Kalise, D., Kröner, A.: Reduced-order minimum time control of advection-reaction-diffusion systems via dynamic programming. In: Proceedings of the 21st International Symposium on Mathematical Theory of Networks and Systems, pp. 1196–1202 (2014)
17. Kunisch, K., Xie, L.: POD-based feedback control of Burgers equation by solving the evolutionary HJB equation. Comput. Math. Appl. **49**, 1113–1126 (2005)
18. Kunisch, K., Volkwein, S., Xie, L.: HJB-POD based feedback design for the optimal control of evolution problems. SIAM J. Appl. Dyn. Syst. **4**, 701–722 (2004)
19. Scherpen, J.: Balancing for nonlinear systems. Syst. Control Lett. **21**, 143–153 (1993)
20. Schmidt, A., Haasdonk, B.: Reduced Basis Approximation of Large Scale Algebraic Riccati Equations. ESAIM: Control, optimisation and Calculus of Variations. EDP Sciences (2017)
21. Sirovich, L.: Turbulence and the dynamics of coherent structures. Parts I-II. Q. Appl. Math. **XVL**, 561–590 (1987)
22. Studinger, A., Volkwein, S.: Numerical analysis of POD a-posteriori error estimation for optimal control. In: Kunisch, K., Bredies, K., Clason, C., von Winckel, G. (eds.) Control and Optimization with PDE Constraints. International Series of Numerical Mathematics, vol. 164, pp. 137–158. Birkhäuser, Basel (2013)
23. Tröltzsch, F.: Optimal Control of Partial Differential Equations: Theory, Methods and Application. American Mathematical Society, Providence (2010)
24. Volkwein, S.: Model reduction using proper orthogonal decomposition. Lecture Notes, University of Konstanz (2013)

Chapter 22
Interpolatory Methods for $\mathscr{H}_\infty$ Model Reduction of Multi-Input/Multi-Output Systems

Alessandro Castagnotto, Christopher Beattie, and Serkan Gugercin

Abstract We develop here a computationally effective approach for producing high-quality $\mathscr{H}_\infty$-approximations to large scale linear dynamical systems having multiple inputs and multiple outputs (MIMO). We extend an approach for $\mathscr{H}_\infty$ model reduction introduced by Flagg et al. (Syst Control Lett 62(7):567–574, 2013) for the single-input/single-output (SISO) setting, which combined ideas originating in interpolatory $\mathscr{H}_2$-optimal model reduction with complex Chebyshev approximation. Retaining this framework, our approach to the MIMO problem has its principal computational cost dominated by (sparse) linear solves, and so it can remain an effective strategy in many large-scale settings. We are able to avoid computationally demanding $\mathscr{H}_\infty$ norm calculations that are normally required to monitor progress within each optimization cycle through the use of "data-driven" rational approximations that are built upon previously computed function samples. Numerical examples are included that illustrate our approach. We produce high fidelity reduced models having consistently better $\mathscr{H}_\infty$ performance than models produced via balanced truncation; these models often are as good as (and occasionally better than) models produced using optimal Hankel norm approximation as well. In all cases considered, the method described here produces reduced models at far lower cost than is possible with either balanced truncation or optimal Hankel norm approximation.

22.1 Introduction

The accurate modeling of dynamical systems often requires that a large number of differential equations describing the evolution of a large number of state variables be integrated over time to predict system behavior. The number of state variables and

A. Castagnotto (✉)
Technical University of Munich, Garching bei München, Germany
e-mail: a.castagnotto@tum.de

C. Beattie • S. Gugercin
Virginia Tech, Blacksburg, VA 24061, USA
e-mail: beattie@vt.edu; gugercin@vt.edu

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_22

differential equations involved can be especially large and forbidding when these models arise, say, from a modified nodal analysis of integrated electronic circuits, or more broadly, from a spatial discretization of partial differential equations over a fine grid. Most dynamical systems arising in practice can be represented at least locally around an operating point, with a state-space representation having the form

$$\begin{aligned} E\dot{x} &= Ax + Bu, \\ y &= Cx + Du, \end{aligned} \tag{22.1}$$

where $E \in \mathbb{R}^{N\times N}$ is the *descriptor matrix*, $A \in \mathbb{R}^{N\times N}$ is the system matrix and $x \in \mathbb{R}^N$, $u \in \mathbb{R}^m$, and $y \in \mathbb{R}^p$ $(p, m \ll N)$ represent the state, input, and output of the system, respectively. A static feed-through relation from the control input u to the control output y is modeled through the matrix $D \in \mathbb{R}^{p\times m}$. Most practical systems involve several actuators (input variables) and several quantities of interest (output variables), motivating our focus here on systems having multiple inputs and multiple outputs (MIMO).

In many application settings, the state dimension N (which typically matches the order of the model) can grow quite large as greater model fidelity is pursued, and in some cases it can reach magnitudes of 10^6 and more. Simulation, optimization, and control design based on such large-scale models becomes computationally very expensive, at times even intractable. This motivates consideration of *reduced order models* (ROMs), which are comparatively low-order models that in spite of having significantly smaller order, $n \ll N$, are designed so as to reproduce the input-output response of the full-order model (FOM) accurately while preserving certain fundamental structural properties, that may include stability and passivity. For state space models such as (22.1), reduced models are obtained generally through Petrov-Galerkin projections having the form:

$$\begin{aligned} \overbrace{W^\top E V}^{E_r}\, \dot{x}_r &= \overbrace{W^\top A V}^{A_r}\, x_r + \overbrace{W^\top B}^{B_r}\, u, \\ y_r &= \underbrace{C V}_{C_r}\, x_r + D_r u. \end{aligned} \tag{22.2}$$

The projection matrices $V, W \in \mathbb{R}^{N\times n}$ become the primary objects of scrutiny in the model reduction enterprise, since how they are chosen has a great impact on the quality of the ROM. For truly large-scale systems, *interpolatory model reduction*, which includes approaches known variously as *moment matching* methods and *Krylov subspace* methods, has drawn significant interest due to its flexibility and comparatively low computational cost [1–3]. Indeed, these methods typically require only the solution of large (generally sparse) linear systems of equations, for which several optimized methods are available. Through the appropriate selection of V and W, it is possible to match the action of the transfer function

$$G(s) = C\,(sE - A)^{-1}\,B + D \tag{22.3}$$

along arbitrarily selected *input* and *output* tangent directions at arbitrarily selected (driving) frequencies. The capacity to do this is central to our approach and is stated briefly here as:

Theorem 1 ([4, 5]) *Let $G(s)$ be the transfer function matrix (22.3) of the FOM (22.1) and let $G_r(s)$ be the transfer function matrix of an associated ROM obtained through Petrov-Galerkin projection as in (22.2). Suppose $\sigma, \mu \in \mathbb{C}$ are complex scalars ("shifts") that do not coincide with any eigenvalues of the matrix pencil (E, A) but otherwise are arbitrary. Let also $r \in \mathbb{C}^m$ and $l \in \mathbb{C}^p$ be arbitrary nontrivial tangent directions. Then*

$$G(\sigma)\cdot r = G_r(\sigma)\cdot r \quad \text{if } (A-\sigma E)^{-1}Br \in \mathsf{Ran}(V), \tag{22.4a}$$

$$l^\top \cdot G(\mu) = l^\top \cdot G_r(\mu) \quad \text{if } (A-\mu E)^{-\top}C^\top l \in \mathsf{Ran}(W), \tag{22.4b}$$

$$l^\top \cdot G'(\sigma)\cdot r = l^\top \cdot G_r'(\sigma)\cdot r \quad \text{if, additionally, } \sigma = \mu. \tag{22.4c}$$

A set of complex shifts, $\{\sigma_i\}_{i=1}^n$, $\{\mu_i\}_{i=1}^n$, with corresponding tangent directions, $\{r_i\}_{i=1}^n$, $\{l_i\}_{i=1}^n$, will be collectively referred to as *interpolation data* in our present context. We define *primitive projection matrices* as

$$\widetilde{V} := \left[(A-\sigma_1 E)^{-1}Br_1, \ldots, (A-\sigma_n E)^{-1}Br_n\right] \tag{22.5a}$$

$$\widetilde{W} := \left[(A-\mu_1 E)^{-\top}C^\top l_1, \ldots, (A-\mu_n E)^{-\top}C^\top l_n\right] \tag{22.5b}$$

Note that $\widetilde{V}$ and $\widetilde{W}$ satisfy Sylvester equations having the form:

$$A\,\widetilde{V} - E\,\widetilde{V}S_\sigma = B\widetilde{R} \quad \text{and} \quad A^\top\widetilde{W} - E^\top\widetilde{W}\,S_\mu^\top = C^\top\widetilde{L}, \tag{22.6}$$

where $S_\sigma = \operatorname{diag}(\sigma_1, .., \sigma_n) \in \mathbb{C}^{n\times n}$, $S_\mu = \operatorname{diag}(\mu_1, .., \mu_n) \in \mathbb{C}^{n\times n}$, $\widetilde{R} = [r_1, .., r_n] \in \mathbb{C}^{m\times n}$ and $\widetilde{L} = [l_1, \ldots, l_n] \in \mathbb{C}^{p\times n}$ [6]. In this way, the Petrov-Galerkin projection of (22.2) is parameterized by interpolation data and the principal task in defining interpolatory models then becomes the judicious choice of shifts and tangent directions.

Procedures have been developed over the past decade for choosing interpolation data that yield reduced models, $G_r(s)$, that minimize, at least locally the approximation error, $G(s) - G_r(s)$, as measured with respect to the $\mathscr{H}_2$-norm:

$$\|G - G_r\|_{\mathscr{H}_2} := \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} \|G(j\omega) - G_r(j\omega)\|_F^2\, d\omega} \tag{22.7}$$

(see [1]). Minimizing the $\mathscr{H}_2$-error, $\|G - G_r\|_{\mathscr{H}_2}$, is of interest through the immediate relationship this quantity bears with the induced system response error:

$$\|y - y_r\|_{\mathscr{L}_\infty} \le \|G - G_r\|_{\mathscr{H}_2}\, \|u(t)\|_{\mathscr{L}_2}, \tag{22.8}$$

A well-known approach to accomplish this that has become popular at least in part due to its simplicity and effectiveness is the *Iterative Rational Krylov Algorithm* (IRKA) [7], which, in effect, runs a simple fixed point iteration aimed at producing interpolation data that satisfy first-order $\mathscr{H}_2$-optimality conditions, i.e.,

$$G(-\lambda_i) \cdot \hat{b}_i = G_r(-\lambda_i) \cdot \hat{b}_i, \quad \hat{c}_i^\top \cdot G(-\lambda_i) = \hat{c}_i^\top \cdot G_r(-\lambda_i), \tag{22.9a}$$

$$\text{and} \quad \hat{c}_i^\top \cdot G'(-\lambda_i) \cdot \hat{b}_i = \hat{c}_i^\top \cdot G_r'(-\lambda_i) \cdot \hat{b}_i. \tag{22.9b}$$

for $i = 1, \ldots, n$. The data λ_i, $\hat{b}_i$ and $\hat{c}_i$ are reduced poles and right/left vector residues corresponding to the pole-residue expansion of the ROM:

$$G_r(s) = \sum_{i=1}^{n} \frac{\hat{c}_i \hat{b}_i^\top}{s - \lambda_i}. \tag{22.10}$$

Despite the relative ease with which $\mathscr{H}_2$-optimal reduced models can be obtained, there are several circumstances in which it might be preferable to obtain a ROM which produces a small error as measured in the $\mathscr{H}_\infty$-norm:

$$\|G - G_r\|_{\mathscr{H}_\infty} := \max_{\omega} \varsigma_{max}(G(j\omega) - G_r(j\omega)), \tag{22.11}$$

where $\varsigma_{max}(M)$ denotes the largest singular value of a matrix M (see [1]). ROMs having small $\mathscr{H}_\infty$-error produce an output response with a uniformly bounded "energy" error:

$$\|y - y_r\|_{\mathscr{L}_2} \leq \|G - G_r\|_{\mathscr{H}_\infty} \|u\|_{\mathscr{L}_2}. \tag{22.12}$$

The $\mathscr{H}_\infty$ norm is also used as a robustness measure for closed-loop control systems and is therefore of central importance in robust control. It finds frequent use in aerospace applications, among others, where the $\mathscr{L}_2$ energy of the system response is of critical interest in design and optimization.

Strategies for producing reduced models that give good $\mathscr{H}_\infty$ performance has long been an active area of research [8]. Analogous to the $\mathscr{H}_\infty$-control design problem, the optimal $\mathscr{H}_\infty$ reduction problem can be formulated in terms of *linear matrix inequalities*, although advantageous features such as linearity and convexity are lost in this case [9, 10]. Due to the high cost related to solving these matrix inequalities, this approach is generally not feasible in large-scale settings.

Another family of methods for the $\mathscr{H}_\infty$ reduction problem relates it to the problem of finding an *optimal Hankel norm approximation* (OHNA) [11–13]. Along these lines the *balanced truncation* (BT) algorithm yields rigorous upper bounds on the $\mathscr{H}_\infty$ error and often produces small approximation error, especially for higher reduced order approximants [1, 14]. Each of these procedures is generally feasible only for mid-size problems since either an all-pass dilation requiring large-scale eigenvalue decomposition (for OHNA) or the solution of generalized Lyapunov equations (for BT) is required. Extensions to large-scale models are available, however—e.g., in [15–20].

A wholly different approach to the $\mathscr{H}_\infty$ model reduction problem for SISO models was proposed by Flagg, Beattie, and Gugercin in [21]. A locally $\mathscr{H}_2$-optimal reduced model is taken as a starting point and adjusted through the variation of rank-one modifications parameterized by the scalar feed-through term, D. Minimization of the $\mathscr{H}_\infty$-error with respect to this parameterization available through D produces ROMs that are observed to have generally very good $\mathscr{H}_\infty$-performance, often exceeding what could be attained with OHNA.

In this work, we extend these earlier interpolatory methods to MIMO systems. We introduce a strategy that reduces the computational expense of the intermediate optimization steps by means of data-driven MOR methods (we use *vector fitting* [22, 23]). Stability of the reduced model is guaranteed through appropriate constraints in the resulting multivariate optimization problem. Numerical examples show effective reduction of approximation error, often outperforming both OHNA and BT.

22.2 MIMO Interpolatory $\mathscr{H}_\infty$-Approximation (MIHA)

In this section we first characterize the $\mathscr{H}_\infty$-optimal reduced order models from the perspective of rational interpolation. This motivates the usage of $\mathscr{H}_2$-optimal reduction as a starting point for the model reduction algorithm we propose for the $\mathscr{H}_\infty$ approximation problem.

22.2.1 Characterization of $\mathscr{H}_\infty$-Approximants via Rational Interpolation

In the SISO case, Trefethen [13] has characterized best $\mathscr{H}_\infty$ approximations within a broader context of rational interpolation:

Theorem 2 (Trefethen [13]) *Suppose $G(s)$ is a (scalar-valued) transfer function associated with a SISO dynamical system as in (22.3). Let $\widehat{G}_r(s)$ be an optimal $\mathscr{H}_\infty$ approximation to $G(s)$ and let G_r be any nth order stable approximation to $G(s)$ that interpolates $G(s)$ at $2n+1$ points in the open right half-plane. Then*

$$\min_{\omega\in\mathbb{R}} |G(j\omega) - G_r(j\omega)| \leq \|G - \widehat{G}_r\|_{\mathscr{H}_\infty} \leq \|G - G_r\|_{\mathscr{H}_\infty}$$

In particular, if $|G(j\omega) - G_r(j\omega)| = \mathsf{const}$ for all $\omega \in \mathbb{R}$ then G_r is itself an optimal $\mathscr{H}_\infty$-approximation to $G(s)$.

For the SISO case, a good $\mathscr{H}_\infty$ approximation will be obtained when the modulus of the error, $|G(s) - G_r(s)|$, is nearly constant as $s = j\omega$ runs along the imaginary axis. In the MIMO case, the analogous argument becomes more technically involved as the maximum singular value of matrix-valued function $G(s) - G_r(s)$ will not generally be analytic in the neighborhood of the imaginary axis (e.g., where multiple singular values occur). Nonetheless, the intuition of the SISO case carries over to the MIMO case, as the following *Gedankenexperiment* might suggest: Suppose that $\widehat{G}_r$ is an $\mathscr{H}_\infty$-optimal interpolatory approximation to G but $\varsigma_{max}(G(j\omega) - G_r(j\omega))$ is not constant with respect to $\omega \in \mathbb{R}$. Then there exist frequencies $\hat{\omega}$ and $\tilde{\omega} \in \mathbb{R}$ and $\epsilon > 0$ such that

$$\begin{aligned}\|G - \widehat{G}_r\|_{\mathscr{H}_\infty} = \varsigma_{max}(G(j\hat{\omega}) - \widehat{G}_r(j\hat{\omega})) &\geq \epsilon + \min_{\omega} \varsigma_{max}(G(j\omega) - \widehat{G}_r(j\omega)) \\ &= \epsilon + \varsigma_{max}(G(j\tilde{\omega}) - \widehat{G}_r(j\tilde{\omega})).\end{aligned}$$

By nudging interpolation data away from the vicinity of $\tilde{\omega}$ and toward $\hat{\omega}$ while simultaneously nudging the poles of $\widehat{G}_r$ away from the vicinity of $\hat{\omega}$ and toward $\tilde{\omega}$, one may decrease the value of $\varsigma_{max}(G(j\hat{\omega}) - \widehat{G}_r(j\hat{\omega}))$ while increasing the value of $\varsigma_{max}(G(j\tilde{\omega}) - \widehat{G}_r(j\tilde{\omega}))$. This will (incrementally) decrease the $\mathscr{H}_\infty$ norm and bring the values of $\varsigma_{max}(G(j\hat{\omega}) - \widehat{G}_r(j\hat{\omega}))$ and $\varsigma_{max}(G(j\tilde{\omega}) - \widehat{G}_r(j\tilde{\omega}))$ closer together toward a common value.

Of course, the nudging process described above contains insufficient detail to suggest an algorithm, and indeed, our approach to this problem follows a somewhat different path, a path that nonetheless uses the guiding heuristic for (near) $\mathscr{H}_\infty$-optimality:

$$\varsigma_{max}(G(j\omega) - \widetilde{G}_r(j\omega)) \approx \mathsf{const} \quad \text{for all } \omega \in \mathbb{R}. \tag{22.13}$$

Approximations with good $\mathscr{H}_\infty$ performance should have an advantageous configuration of poles and interpolation data that locates them symmetrically about the imaginary axis, thus balancing regions where $\varsigma_{max}(G(s) - \widetilde{G}_r(s))$ is big (e.g., pole locations) symmetrically against regions reflected across the imaginary axis where $\varsigma_{max}(G(s) - \widetilde{G}_r(s))$ is small (e.g., interpolation locations). This configuration of poles and interpolation data, we note, is precisely the outcome of optimal $\mathscr{H}_2$ approximation as well, and this will provide us with an easily computable approximation that is likely to have good $\mathscr{H}_\infty$ performance.

22.2.2 $\mathscr{H}_\infty$ Approximation with Interpolatory $\mathscr{H}_2$-Optimal Initialization

Local $\mathscr{H}_2$-optimal ROMs are often observed to give good $\mathscr{H}_\infty$ performance—this is in addition to the expected good $\mathscr{H}_2$ performance. This $\mathscr{H}_\infty$ behaviour is illustrated in Fig. 22.1, where the $\mathscr{H}_\infty$ approximation errors of local $\mathscr{H}_2$-optimal ROMs produced by IRKA are compared to ROMs of the same order obtained through BT for the CD player MIMO benchmark model [24].

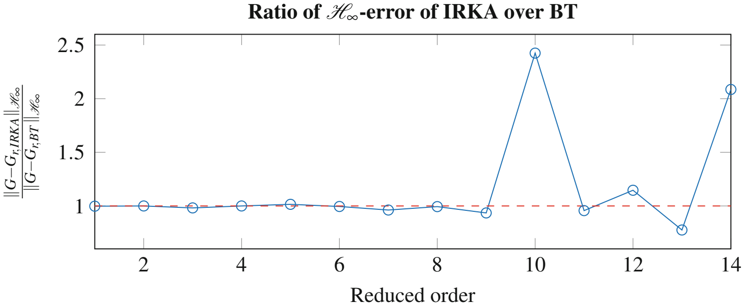


Fig. 22.1 Numerical investigations indicate that IRKA models are often good also in terms of the $\mathscr{H}_\infty$-error

The frequently favourable $\mathscr{H}_\infty$ behaviour of IRKA models has particular significance in this context, since they are computationally cheap to obtain even in large-scale settings, indeed often they are much cheaper than comparable BT computations. The resulting locally $\mathscr{H}_2$-optimal ROMs can be further improved (with respect to $\mathscr{H}_\infty$ error) by relaxing the (implicit) interpolation constraint at ∞ while preserving the $\mathscr{H}_2$-optimal interpolation conditions (which is the most important link the $\mathscr{H}_2$-optimal ROM has with the original model).

Consider the partial fraction expansion

$$G_r(s) = \sum_{i=1}^{N} \frac{\hat{c}_i \hat{b}_i^\top}{s-\lambda_i} + D_r. \tag{22.14}$$

For ease of exposition, we assume the poles, λ_i, to be simple, although the results we develop here can be extended to the case of higher multiplicity. The input/output behavior is determined by n scalar parameters λ_i, n pairs of input/output residuals $\hat{b}_i$, $\hat{c}_i$ and the $p \times m$-dimensional feed-through D_r. Considering that a constant scaling factor can be arbitrarily defined in the product of the residuals, this leaves us a total of $n\,(p+m) + p \cdot m$ parameters, $n\,(p+m)$ of which can be described in terms of two-sided tangential interpolation conditions (22.4). This interpolation data is established for the original $\mathscr{H}_2$-optimal ROM and we wish it to remain invariant over subsequent adjustments, so the only remaining degrees-of-freedom are the $p \cdot m$ entries in the feed-through matrix D_r.

In the typical context of $\mathscr{H}_2$-optimal model reduction, D_r is chosen to match the feed-through term D of the original model, thus guaranteeing that the error $G - \widetilde{G}_r$ remains in $\mathscr{H}_2$. Note that D remains untouched by the state-space projections in (22.2), moreover since typically $p, m \ll N$, the feed-through term need not be involved in the reduction process and may be retained from the FOM. Indeed, retaining the original feed-through term is a necessary condition for $\mathscr{H}_2$ optimality, forcing interpolation at $s = \infty$ and as a consequence, small error at higher frequencies. Contrasting significantly with $\mathscr{H}_2$-based model reduction, good $\mathscr{H}_\infty$ performance does not require $D_r = D$, and in this work we exploit this flexibility in a crucial way. A key observation playing a significant role in what follows was made in [25, 26] that the feed-through term D_r induces a parametrization of all reduced order models satisfying the two-sided tangential interpolation conditions. This result is summarized by following theorem taken from [25, Theorem 4.1] and [26, Theorem 3]

Theorem 3 *Let $\widetilde{R}$, $\widetilde{L}$ be defined through the Sylvester equations in (22.6). Assume, without loss of generality, that the full order model satisfies $D = 0$ and let the nominal reduced model $G_r^0(s) = C_r\,(sE_r - A_r)^{-1}\,B_r$ be obtained through Petrov-Galerkin projection using the primitive projection matrices (22.5). Then, for any $D_r \in \mathbb{C}^{p\times m}$, the perturbed reduced order model*

$$\widetilde{G}_r^D(s, D_r) = \left(\widetilde{C}_r + D_r\widetilde{R}\right)\left[s\widetilde{E}_r - \left(\widetilde{A}_r + \widetilde{L}^\top D_r\widetilde{R}\right)\right]^{-1}\left(\widetilde{B}_r + \widetilde{L}^\top D_r\right) + D_r \tag{22.15}$$

also satisfies the tangential interpolation conditions (22.4).

Note that for $D \neq 0$, the results of Theorem 3 can be trivially extended by adding D to the right-hand side in (22.15). Even though for theoretical consideration the use of primitive Krylov bases $\widetilde{V}$, $\widetilde{W}$ introduced in (22.5) is often convenient, from a numerical standpoint there are several reason why one may choose a different basis for the projection matrices. This next result shows that the interpolation conditions are preserved also for arbitrary bases—in particular also real and orthonormal bases—provided that the shifting matrices R and L are appropriately chosen.

Corollary 1 *Let $T_v, T_w \in \mathbb{C}^{n\times n}$ be invertible matrices used to transform the primitive bases $\widetilde{V}$, $\widetilde{W}$ of the Krylov subspace to new bases $V = \widetilde{V}T_v$ and $W = \widetilde{W}T_w$. Let the same transformation be applied to the matrices of tangential directions, resulting in $R = \widetilde{R}T_v$ and $L = \widetilde{L}T_w$. Then, for any D_r, the ROM G_r^D is given by*

$$G_r^D(s, D_r) = \underbrace{(C_r + D_rR)}_{C_r^D}\left[sE_r - \underbrace{\left(A_r + L^\top D_rR\right)}_{A_r^D}\right]^{-1}\underbrace{\left(B_r + L^\top D_r\right)}_{B_r^D} + D_r \tag{22.16}$$

Proof The proof amounts to showing that the transfer function matrix G_r^D of the ROM is invariant to a change of basis from $\widetilde{V}$ and $\widetilde{W}$ as long as $\widetilde{R}$ and $\widetilde{L}$ are adapted accordingly.

$$\begin{aligned}
&G_r^D - D_r = C_r^D\left(sE - A_r^D\right)^{-1}B_r^D\\
&= (CV + D_rR)\left[sW^\top EV - W^\top AV - L^\top D_rR^\top\right]^{-1}\left(W^\top B + L^\top D_r\right)\\
&= \left(C\widetilde{V} + D_r\widetilde{R}\right)T_v\left[T_w^\top\left(s\widetilde{W}^\top E\widetilde{V} - \widetilde{W}^\top A\widetilde{V} - \widetilde{L}^\top D_r\widetilde{R}^\top\right)T_v\right]^{-1}T_w^\top\left(\widetilde{W}^\top B + \widetilde{L}^\top D_r\right)\\
&= \left(C\widetilde{V} + D_r\widetilde{R}\right)\left[s\widetilde{W}^\top E\widetilde{V} - \widetilde{W}^\top A\widetilde{V} - \widetilde{L}^\top D_r\widetilde{R}^\top\right]^{-1}\left(\widetilde{W}^\top B + \widetilde{L}^\top D_r\right)\\
&= \widetilde{G}_r^D - D_r\,.
\end{aligned}$$

The results of Theorem 3 generalize to the case of arbitrary bases. Following the notation from [25, Definition 2.1], the state-space models resulting from Petrov-Galerkin projections with V, W and $\widetilde{V}, \widetilde{W}$ respectively are *restricted system equivalent*. As a consequence, they share the same transfer function matrix.

Using the Sherman-Morrison-Woodbury formula [27] for the inverse of rank k perturbations of a matrix, we are able to decompose the transfer function of the shifted reduced model into the original reduced model and an additional term.

Corollary 2 *Define the auxiliary variable $\mathcal{K}_r := sE_r - A_r$. The transfer function of the shifted reduced model G_r^D can be given as*

$$G_r^D(s) = G_r^0(s) + \Delta G_r^D(s, D_r), \tag{22.17}$$

where G_r^0 is the transfer function of the unperturbed model and ΔG_r^D is defined as

$$\begin{gathered}\Delta G_r^D = \Delta_1 + \Delta_2 + \Delta_3 \cdot (\Delta_4)^{-1} \cdot \Delta_2 + D_r \\ \textit{given} \\ \begin{aligned} \Delta_1 &:= C_r \mathcal{K}_r^{-1} L^\top D_r & \Delta_2 &:= D_r R \mathcal{K}_r^{-1} \left(B_r + L^\top D_r\right) \\ \Delta_3 &:= \left(C_r + D_r R\right) \mathcal{K}_r^{-1} L^\top & \Delta_4 &:= I - D_r R \mathcal{K}_r^{-1} L^\top \end{aligned}\end{gathered} \tag{22.18}$$

Proof Note that by the Sherman-Morrison-Woodbury formula, following equality holds:

$$\left(\mathcal{K}_r - L^\top D_r R\right)^{-1} = \mathcal{K}_r^{-1} + \mathcal{K}_r^{-1} L^\top \left(I - D_r R \mathcal{K}_r^{-1} L^\top\right)^{-1} D_r R \mathcal{K}_r^{-1}. \tag{22.19}$$

Using this relation in the definition of G_r^D, the proof is completed by straightforward algebraic manipulations.

We proceed by attempting to exploit the additional degrees-of-freedom available in D_r to trade off excessive accuracy at high frequencies for improved approximation in lower frequency ranges, as measured with the $\mathcal{H}_\infty$-norm. We first obtain an $\mathcal{H}_2$-optimal ROM by means of IRKA and subsequently minimize the $\mathcal{H}_\infty$-error norm with respect to the constant feed-through matrix D_r while preserving tangential interpolation and guaranteeing stability. The resulting ROM G_r^* will represent a local optimum out of the set of all stable ROMs satisfying the tangential interpolation conditions. The outline of our proposed reduction procedure, called *MIMO interpolatory $\mathcal{H}_\infty$-approximation* (MIHA), is given in Algorithm 1.

Numerical results in Sect. 22.3 will show the effectiveness of this procedure in further reducing the $\mathcal{H}_\infty$-error for a given IRKA model. However, at this stage the optimization in Step 2 appears problematic, for it requires both the computation of the $\mathcal{H}_\infty$-norm of a large-scale model and a constrained multivariate optimization of a non-convex, non-smooth function. It turns out that both of these issues can be resolved effectively, as it will be discussed in the following sections.

Algorithm 1 MIMO interpolatory $\mathscr{H}_\infty$-approximation (MIHA)

```
Input: $G(s)$, $n$
Output: Stable, locally optimal reduced order model $G_r^*$, approximation error $e^*_{\mathscr{H}_\infty}$
1: $G_r^0 \leftarrow \text{IRKA}(G(s), n)$
2: $D_r^* \leftarrow \arg\min_{D_r} \left\| G(s) - G_r^D(s, D_r) \right\|_{\mathscr{H}_\infty}$ s.t. $G_r^D(s, D_r^*)$ is stable
3: $G_r^* \leftarrow G_r^D(s, D_r^*)$
4: $e^*_{\mathscr{H}_\infty} \leftarrow \left\| G(s) - G_r^*(s) \right\|_{\mathscr{H}_\infty}$
```

22.2.3 *Efficient Implementation*

As we have noted, the main computational burden of the algorithm described above resides mainly in Step 2. We are able to lighten this burden somewhat through judicious use of (22.17) and by taking advantage of previously computed transfer function evaluations.

22.2.3.1 A "Free" Surrogate Model for the Approximation Error $G - G_r^0$

Step 1 of Algorithm 1 requires performing $\mathscr{H}_2$-optimal reduction using IRKA. This is a fixed point iteration involving a number of steps k before convergence is achieved. At every step j, Hermite tangential interpolation about some complex frequencies $\{\sigma_i\}_{i=1}^n$ and tangential directions $\{r_i\}_{i=1}^n$, $\{l_i\}_{i=1}^n$ is performed. For this purpose, the projection matrices in (22.5) are computed, and it is easy to see that for all $i = 1, \ldots, n$ it holds

$$C \cdot \widetilde{V} e_i = C \left(A - \sigma_i E\right)^{-1} B r_i = G(\sigma_i) r_i \tag{22.20a}$$

$$e_i^\top \widetilde{W}^\top \cdot B = l_i^\top C \left(A - \sigma_i E\right)^{-1} B = l_i^\top G(\sigma_i) \tag{22.20b}$$

$$e_i^\top \widetilde{W}^\top E \widetilde{V} e_i = l_i^\top \left(A - \sigma_i E\right)^{-1} E \left(A - \sigma_i E\right)^{-1} r_i = l_i^\top G'(\sigma_i) r_i \tag{22.20c}$$

Observe that, at basically no additional cost, we can gather information about the FOM while performing IRKA. Figure 22.2a illustrates this point by showing the development of the shifts during the IRKA iterations reducing the CDplayer benchmark model to a reduced order $n = 10$. For all complex frequencies indicated by a marker, tangent data for the full order model is collected.

To use this "free" data, there are various choices for "data-driven" procedures that produce useful rational approximations. *Loewner methods* [25, 28–30] are effective and are already integrated into IRKA iteration strategies [31]. We adopt here a *vector fitting* strategy [22, 23, 32–34] instead. This allows us to produce stable low-order approximations of the reduction error after IRKA

$$\widetilde{G_e^0} \approx G_e^0 := G - G_r^0. \tag{22.21}$$

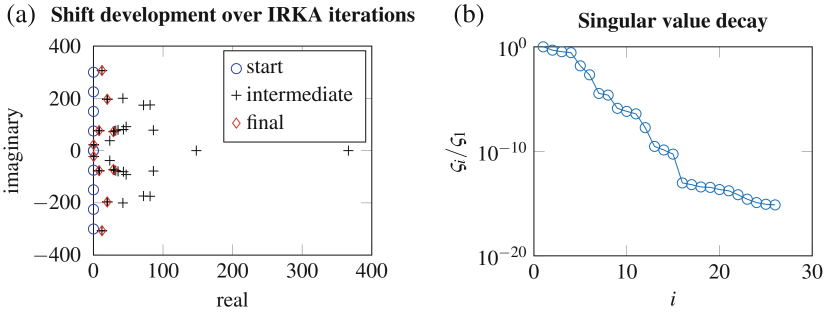


Fig. 22.2 Data collecting during IRKA can be used to generate data-driven surrogates. (**a**) Points at which data of the FOM is collected during IRKA. (**b**) Decay of singular values of the matrix $[\mathbb{L}, \sigma\mathbb{L}]$ for the data collected during IRKA

An appropriate choice of order for the surrogate model can be obtained by forming the Loewner $\mathbb{L}$ and shifted Loewner $\sigma\mathbb{L}$ matrices from G and G' evaluations that were generated in the course of the IRKA iteration and then observing the singular value decay of the matrix $[\mathbb{L}, \sigma\mathbb{L}]$, as indicated in Fig. 22.2b.

Using the decomposition in (22.17), the $\mathcal{H}_\infty$-norm evaluations required during the optimization will be feasible even for large-scale full order models. In addition, it will allow us to obtain a cheap estimate $\tilde{e}_{\mathcal{H}_\infty}$ for the approximation error

$$e_{\mathcal{H}_\infty} := \left\|G - G_r^D\right\|_{\mathcal{H}_\infty} \approx \left\|\widetilde{G_e^0} - \Delta G_r^D\right\|_{\mathcal{H}_\infty} = \tilde{e}_{\mathcal{H}_\infty} \qquad (22.22)$$

22.2.3.2 Constrained Multivariate Optimization with Respect to D_r

The focus of this work lies in the development of new model reduction strategies. Our intent is not directed toward making a contribution to either the theory or practice of numerical optimization and we are content in this work to use standard optimization approaches. In the results of Sect. 22.3, we rely on state-of-the-art algorithms that are widespread and available, e.g., in MATLAB. With that caveat understood, we do note that the constrained multivariate optimization over the reduced feed-through, D_r, is a challenging optimization problem, so we will explain briefly the setting that seems to work best in our case. The computation and optimization of $\mathcal{H}_\infty$-norms for large-scale models remains an active area of research, as demonstrated by Mitchell and Overton [35, 36] and Aliyev et al. [37].

The problem we need to solve in step 2 of Algorithm 1 is

$$\begin{aligned} &\min_{D_r \in \mathbb{R}^{p\times m}} \max_{\omega} \varsigma_{max}\left(G(j\omega) - G_r^D(j\omega, D_r)\right) \\ &s.t. \quad G_r^D(s, D_r) \text{ is stable} \end{aligned} \qquad (22.23)$$

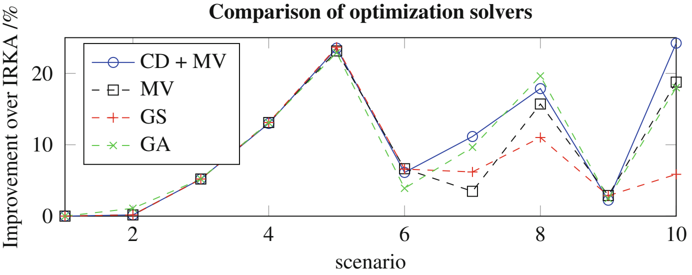


Fig. 22.3 Comparison of different solvers shows the effectiveness of coordinate descent followed by multivariate optimization

which represents a non-smooth, non-convex multivariate optimization problem in a $p \times m$-dimensional search space. In our experience, the best strategy considering both optimization time and optimal solution is given by a combination of *coordinate descent* (CD) [38] and subsequent *multivariate optimization* (MV). We refer to this combined strategy as CD+MV. The coordinate descent strategy is used in this setting somewhat like an initialization procedure to find a better starting point than $D_r^0 = 0$. This initialization is based on reducing the search space from $p \cdot m$ dimensions to just one, hence performing a much simpler univariate optimization in each step. Once one cycle has been conducted for all elements in the feed-through matrix, the resulting feed-through is used to initialize a nonlinear constrained optimization solver that minimizes the error with respect to the whole D_r matrix. We have used a sequential quadratic programming (SQP) method as implemented in MATLAB's `fmincon`, although acceptable options for this final step abound. Further information about optimization strategies can be found in [39].

The suitability of CD+MV is motivated by extensive simulations conducted comparing different strategies, such as direct multivariate optimization, *global search* (GS) [40], and *genetic algorithms* (GA) (cp. Fig. 22.3). Ultimately, we rely on the results of Sect. 22.3 to show that this procedure is effective.

22.3 Numerical Results

In the following we demonstrate the effectiveness of the proposed procedure by showing reduction results with different MIMO models. The reduction code is based on the sssMOR toolbox[1] [41]. For generation of vector fitting surrogates, we use the `vectfit3` function[2] [22, 32, 33]. Note that more recent implementation of MIMO

[1] Available at www.rt.mw.tum.de/?sssMOR.

[2] Available at www.sintef.no/projectweb/vectfit/downloads/vfut3/.

vector fitting introduced in [23] could be used instead, especially for improved robustness.

22.3.1 Heat Model

Our proposed procedure is demonstrated through numerical examples conducted on a MIMO benchmark model representing a discretized heat equation of order $N = 197$ with $p = 2$ outputs and $m = 2$ inputs [42].

Model reduction for this model was conducted for a range of reduced orders; the results are summarized in Table 22.1. The table shows the reduced order n, the order n_m of the error surrogate $\widetilde{G}_e^0$, and the relative $\mathcal{H}_\infty$ error of the proposed ROM G_r^D, as well as the percentage improvement over the initial IRKA model. Our proposed method improves significantly on the $\mathcal{H}_\infty$ performance of IRKA, in some cases by more than 50%.

Figure 22.4 gives a graphical representation of the reduction results. The plots compare the approximation error achieved after applying MIHA, with a vector fitting surrogate as described in Sect. 22.2.3.1, to other reduction strategies. These include the direct reduction with IRKA, balanced truncation (BT), Optimal Hankel Norm Approximation (OHNA) as well as the optimization with respect to the actual error G_e^0 (MIHA without surrogate). For a better graphical comparison throughout the reduced orders studied, the errors are related to the theoretical lower bound given by

$$\underline{e}_{\mathcal{H}_\infty} := \varsigma_{n+1}^H, \tag{22.24}$$

Table 22.1 Results for the heat model problem

n	1	2	3	4	5	6	7	8	9	10
n_m	14	24	20	22	24	30	32	36	36	36
$\frac{\|G-G_r^D\|}{\|G\|}$	8.7e-2	7.6e-3	1.2e-2	1.2e-3	6.5e-4	5.7e-4	4.1e-4	1.6e-4	4.4e-5	8.6e-6
$1-\frac{\|G-G_r^D\|}{\|G-G_r^0\|}$	50.8%	39.0%	27.0%	36.7%	36.0%	44.8%	52.0%	44.6%	49.5%	42.6%

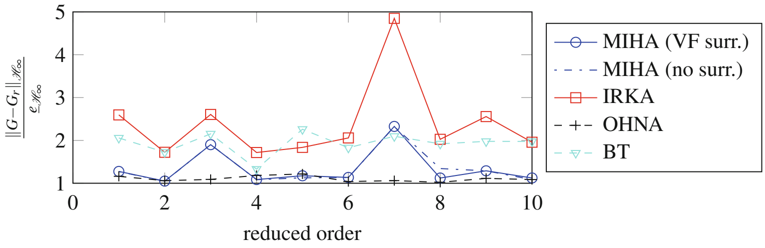


Fig. 22.4 Plot of the approximation error relative to the theoretical error bound

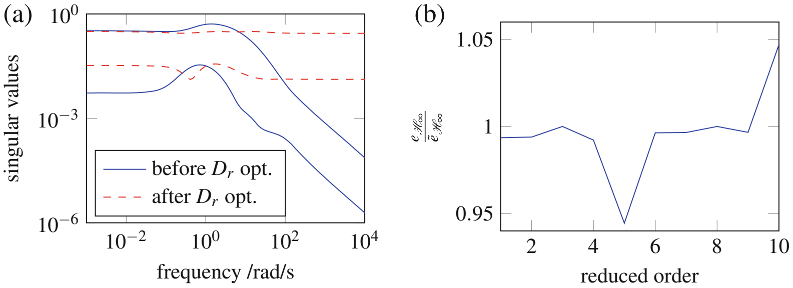


Fig. 22.5 Optimization with the surrogate effectively reduces and provides an accurate estimate of the true error. (**a**) Singular value plot of the error before and after optimization. (**b**) Comparison of error estimate $\tilde{e}_{\mathcal{H}_\infty}$ versus true error $e_{\mathcal{H}_\infty}$

with which we denote the Hankel singular value of order $n + 1$.

Notice how effectively the ROMs resulting from the D_r-optimization reduce the $\mathcal{H}_\infty$-error beyond what is produced by the IRKA ROMs and that they often, (here, in 9 out of 10 cases) yield better results than BT and sometimes (here, in 3 out of 10 cases) yield better results even than OHNA. Note also that the optimization with respect to the vector-fitting surrogate produces as good a result as optimization with respect to the true error. For reduced order $n = 8$, optimization with respect to the surrogate yields even a better result. This is not expected and may be due to the different cost functions involved, causing optimization of the true error to converge to a worse solution.

The plot also confirms our initial motivation in using IRKA models as starting points, since their approximation in terms of the $\mathcal{H}_\infty$ norm is often not far from BT. Finally, note how in several cases the resulting ROM is very close to the theoretical lower bound, which implies that the respective ROMs are not far from being the *global* optimum.

Figure 22.5a shows the approximation error before and after the feed-through optimization for a selected reduced order of 2. The largest singular value is drastically reduced (ca. 40%) by lifting up the value at high frequencies. This confirms our intuition that the $\mathcal{H}_\infty$-optimal reduced order model should have a nearly constant error modulus over all frequencies. Finally, Fig. 22.5b demonstrates the validity of the error estimate $\tilde{e}_{\mathcal{H}_\infty}$ obtained using the surrogate model.

22.3.2 *ISS Model*

Similar simulations were conducted on a MIMO model with $m = 3$ inputs and $p = 3$ outputs of order $N = 270$, representing the 1r component of the International Space Station (ISS) [24]. The results are summarized in Table 22.2 and Fig. 22.6. Note that the $\mathcal{H}_\infty$-error after IRKA is comparable to that of BT and the proposed

Table 22.2 Results for the ISS problem

n	2	4	6	8	10	12	14	16	18	20
n_m	12	18	12	18	18	15	42	48	30	30
$\frac{\|G-G_r^D\|}{\|G\|}$	2.7e-1	9.4e-2	8.4e-2	7.9e-2	3.6e-2	3.4e-2	2.2e-2	2.2e-2	1.0e-2	7.7e-3
$1-\frac{\|G-G_r^D\|}{\|G-G_r^0\|}$	7.5%	9.9%	8.8%	4.9%	9.5%	13.8%	23.3%	15.7%	3.5%	25.8%

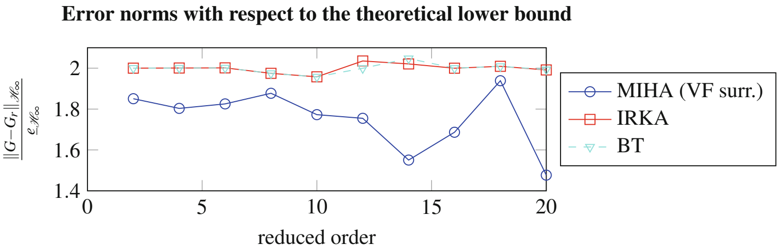


Fig. 22.6 Plot of the approximation error relative to the theoretical error bound (ISS)

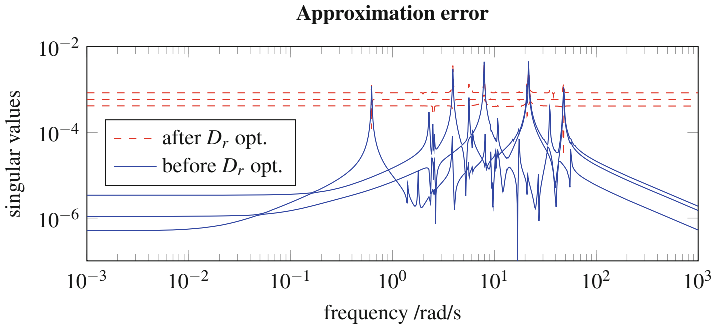


Fig. 22.7 Singular value plot of the error before and after optimization (ISS)

procedure is effective in further reducing the error, outperforming BT in all cases investigated.

Finally, note also in this case that the modulus of the error due to this $\mathcal{H}_\infty$-approximation procedure is nearly constant, as anticipated. This is demonstrated in Fig. 22.7, where the error plots for the reduction order $n = 10$ are compared.

Acknowledgements The work of the first author was supported by the German Research Foundation (DFG), Grant LO408/19-1, as well as the TUM Fakultäts-Graduiertenzentrum Maschinenwesen; the work of the second author was supported by the Einstein Foundation, Berlin; the work of the third author was supported by the Alexander von Humboldt Foundation.

References

1. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. Society for Industrial and Applied Mathematics, Philadelphia, PA (2005)
2. Gallivan, K.A., Vandendorpe, A., Van Dooren, P.: On the generality of multipoint padé approximations. In: 15th IFAC World Congress on Automatic Control (2002)
3. Beattie, C.A., Gugercin, S.: Model reduction by rational interpolation. In: Benner, P., Cohen, A., Ohlberger, M., Willcox, K. (eds.) Model Reduction and Approximation: Theory and Algorithms. SIAM, Philadelphia (2017)
4. Grimme, E.J.: Krylov Projection Methods for Model Reduction. Ph.D. thesis, Department of Electrical Engineering, University of Illinois at Urbana Champaign, 1997
5. Gallivan, K.A., Vandendorpe, A., Van Dooren, P.: Model reduction of MIMO systems via tangential interpolation. SIAM J. Matrix Anal. Appl. **26**(2), 328–349 (2004)
6. Gallivan, K.A., Vandendorpe, A., Van Dooren, P.: Sylvester equations and projection-based model reduction. J. Comput. Appl. Math. **162**(1), 213–229 (2004)
7. Gugercin, S., Antoulas, A.C., Beattie, C.A.: $\mathscr{H}_2$ model reduction for large-scale linear dynamical systems. SIAM J. Matrix Anal. Appl. **30**(2), 609–638 (2008)
8. Antoulas, A.C., Astolfi, A.: $\mathscr{H}_\infty$-norm approximation. In: V. Blondel, A. Megretski (eds.) Unsolved Problems in Mathematical Systems and Control Theory, pp. 267–270. Princeton University Press, Princeton (2002)
9. Helmersson, A.: Model reduction using LMIs. In: Proceedings of 1994 33rd IEEE Conference on Decision and Control, vol. 4, pp. 3217–3222. Lake Buena Vista, IEEE (1994). doi:10.1109/CDC.1994.411635
10. Varga, A., Parrilo, P.: Fast algorithms for solving $\mathscr{H}_\infty$-norm minimization problems. In: Proceedings of the 40th IEEE Conference on Decision and Control, 2001, vol. 1, pp. 261–266. IEEE, Piscataway, NJ (2001)
11. Kavranoglu, D., Bettayeb, M.: Characterization of the solution to the optimal $\mathscr{H}_\infty$ model reduction problem. Syst. Control Lett. **20**(2), 99–107 (1993)
12. Glover, K.: All optimal hankel-norm approximations of linear multivariable systems and their linf-error bounds. Int. J. Control **39**(6), 1115–1193 (1984)
13. L.N. Trefethen, Rational Chebyshev approximation on the unit disk. Numer. Math. **37**(2), 297–320 (1981)
14. Gugercin, S., Antoulas, A.C.: A survey of model reduction by balanced truncation and some new results. Int. J. Control **77**(8), 748–766 (2004)
15. Benner, P., Quintana-Ortí, E.S., Quintana-Ortí, G.: Computing optimal Hankel norm approximations of large-scale systems. In: 43rd IEEE Conference on Decision and Control, 2004. CDC, vol. 3, pp. 3078–3083. IEEE, Piscataway, NJ (2004)
16. Li, J.-R.: Model reduction of large linear systems via low rank system gramians, Ph.D. thesis, Massachusetts Institute of Technology, 2000
17. Benner, P., Kürschner, P., Saak, J.: Self-generating and efficient shift parameters in ADI methods for large Lyapunov and Sylvester equations. Electron. Trans. Numer. Anal. **43**, 142–162 (2014)
18. Kürschner, P.: Efficient low-rank solution of large-scale matrix equations, Ph.D. thesis, Otto-von-Guericke Universität Magdeburg (2016)
19. Sabino, J.: Solution of large-scale Lyapunov equations via the block modified Smith method. Ph.D. thesis, Citeseer (2006)
20. Gugercin, S., Sorensen, D.C., Antoulas, A.C.: A modified low-rank smith method for large-scale Lyapunov equations. Numerical Algoritm. **32**(1), 27–55 (2003)
21. Flagg, G.M., Beattie, C.A., Gugercin, S.: Interpolatory $\mathscr{H}_\infty$ model reduction. Syst. Control Lett. **62**(7), 567–574 (2013)
22. Gustavsen, B., Semlyen, A.: Rational approximation of frequency domain responses by vector fitting. IEEE Trans. Power Delivery **14**(3), 1052–1061 (1999)

23. Drmač, Z., Gugercin, S., Beattie, C.: Vector fitting for matrix-valued rational approximation. SIAM J. Sci. Comput. **37**(5), A2346–A2379 (2015)
24. Chahlaoui, Y., Van Dooren, P.: A collection of benchmark examples for model reduction of linear time invariant dynamical systems. Working Note 2002-2, 2002
25. Mayo, A.J., Antoulas, A.C.: A framework for the solution of the generalized realization problem. Linear Algebra Appl. **425**(2–3), 634–662 (2007)
26. Beattie, C.A., Gugercin, S.: Interpolatory projection methods for structure-preserving model reduction. Syst. Control Lett. **58**(3), 225–232 (2009)
27. Golub, G.H., Van Loan, C.F.: Matrix Computations, vol. 3. JHU Press, Baltimore (2012)
28. Antoulas, A.C., Anderson, B.D.Q.: On the scalar rational interpolation problem. IMA J. Math. Control Inf. **3**(2-3), 61–88 (1986)
29. Anderson, B.D.O., Antoulas, A.C.: Rational interpolation and state-variable realizations. Linear Algebra Appl. **137**, 479–509 (1990)
30. Lefteriu, S., Antoulas, A.C.: A new approach to modeling multiport systems from frequency-domain data. IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. **29**(1), 14–27 (2010)
31. Beattie, C.A., Gugercin, S.: Realization-independent $\mathscr{H}_2$-approximation. In: 51st IEEE Conference on Decision and Control, pp. 4953–4958. IEEE, Piscataway, NJ (2012)
32. Gustavsen, B.: Improving the pole relocating properties of vector fitting. IEEE Trans. Power Delivery **21**(3), 1587–1592 (2006)
33. Deschrijver, D., Mrozowski, M., Dhaene, T., De Zutter, D.: Macromodeling of multiport systems using a fast implementation of the vector fitting method. IEEE Microwave Wireless Compon. Lett. **18**(6), 383–385 (2008)
34. Drmac, Z., Gugercin, S., Beattie, C.: Quadrature-based vector fitting for discretized h_2 approximation. SIAM J. Sci. Comput. **37**(2), A625–A652 (2015)
35. Mitchell, T., Overton, M.L.: Fixed low-order controller design and $\mathscr{H}_\infty$ optimization for large-scale dynamical systems. IFAC-PapersOnLine **48**(14), 25–30 (2015)
36. Mitchell, T., Overton, M.L.: Hybrid expansion–contraction: a robust scaleable method for approximating the $\mathscr{H}_\infty$ norm. IMA J. Numer. Anal. **36**(3), 985–1014 (2016)
37. Aliyev, N., Benner, P., Mengi, E., Voigt, M.: Large-scale computation of $\mathscr{H}_\infty$ norms by a greedy subspace method. SIAM J. Matrix Anal. Appl. (2016, submitted to). http://portal.ku.edu.tr/~emengi/papers/large_hinfinity.pdf
38. Wright, S.J.: Coordinate descent algorithms. Math. Program. **151**(1), 3–34 (2015)
39. Nocedal, J., Wright, S.: Numerical Optimization. Springer Science and Business Media, New York (2006)
40. Ugray, Z., Lasdon, L., Plummer, J., Glover, F., Kelly, J., Martí, R.: Scatter search and local NLP solvers: a multistart framework for global optimization. INFORMS J. Comput. **19**(3), 328–340 (2007)
41. Castagnotto, A., Varona, M.C., Jeschek, L., Lohmann, B.: sss and sssMOR: Analysis and reduction of large-scale dynamic systems with MATLAB. at-Automatisierungstechnik **65**(2), 134–150(2017). doi:10.1515/auto-2016-0137
42. Antoulas, A.C., Sorensen, D.C., Gugercin, S.: A survey of model reduction methods for large-scale systems. In: Structured Matrices in Opera Structured, Numerical Analysis, Control, Signal and Image Processing. Contemporary Mathematics, vol. 280, pp. 193–219. AMS Publications, Providence, RI (2001)

Chapter 23
Model Reduction of Linear Time-Varying Systems with Applications for Moving Loads

M. Cruz Varona and B. Lohmann

Abstract In this paper we consider different model reduction techniques for systems with moving loads. Due to the time-dependency of the input and output matrices, the application of time-varying projection matrices for the reduction offers new degrees of freedom, which also come along with some challenges. This paper deals with both, simple methods for the reduction of particular linear time-varying systems, as well as with a more advanced technique considering the emerging time derivatives.

23.1 Introduction

The detailed modeling of physical and technical phenomena arising in many engineering and computer science applications may yield models of very large dimension. This is particular the case in fields such as thermo-fluid dynamics, structural mechanics or integrated circuit design, where the models are mostly obtained from a spatial discretization of the underlying partial differential equations. The resulting large systems of ordinary differential equations or differential-algebraic equations are computationally expensive to simulate and handle. In order to reduce the computational effort, model reduction techniques that generate reduced-order models that approximate the dynamic behaviour and preserve the relevant properties of the original model are required. For the reduction of linear time-invariant (LTI) systems, various well-established reduction approaches exist (see e.g. [2]). In the past 10 years, further model reduction methods have been developed for linear, parametric and nonlinear systems [19, 6, 17, 5] and applied in a wide variety of domains.

In this contribution, we investigate model order reduction of linear time-varying (LTV) systems. Such systems arise in many real-life applications, since dynamical systems often depend on parameters which vary over time or might alter their behaviour due to ageing, degradation, environmental changes and time-dependent

M. Cruz Varona (✉) • B. Lohmann
Chair of Automatic Control, Technical University of Munich, 85748 Garching, München, Germany
e-mail: maria.cruz@tum.de; lohmann@tum.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_23

operating conditions. Another possible application for LTV systems are moving loads. This particular but still very frequent problem arises, for example, in working gears, cableways, bridges with moving vehicles or milling processes. Since the position of the acting force varies over time, systems with sliding components exhibit a time-variant behaviour. The varying load location can be modeled and considered in different ways, thus yielding diverse alternative representations for systems with moving loads and, according to this, leading to different approaches to reduce them.

One possibility is to represent moving loads as LTV systems, in which *only* the input and/or output matrices $\mathbf{B}(t)$ and $\mathbf{C}(t)$ are time-dependent. Such systems can be then reduced using balanced truncation model reduction methods developed in [20, 18]. These approaches, however, require a high computational and storage effort, since two differential Lyapunov equations must be solved. Recently, a practical and efficient procedure of balanced truncation for LTV systems has been presented in [14]. Note that these aforementioned balanced truncation techniques can be applied to general LTV systems, where all system matrices are time-dependent. For the reduction of systems with only time-varying input and output matrices the two-step approach proposed in [3, 21] can also be pursued. This method consists first on a low-rank approximation of the time-dependent input matrix and consequently on applying standard model reduction techniques to the resulting LTI system with a modified input. The approximation of the input matrix in a low-dimensional subspace is performed via the solution of a linear least squares minimization problem.

Systems with moving loads can further be modeled by means of linear switched systems. Well-known reduction methods such as balanced truncation can then be applied for the reduction of each LTI subsystem [13].

A last alternative option for describing systems with moving loads is to consider the load position as a time-dependent parameter of the system model. This results in a linear parameter-varying (LPV) system, in which only the input and/or output matrices depend on a time-varying parameter. In many recent publications, e.g. [3, 10, 11, 13], the parameter is assumed to be time-independent. Thereby any parametric model order reduction (pMOR) approach [1, 4, 6, 16] can be applied to the resulting parametric LTI system. In some other recent publications [7–9, 22], however, the time variation of the parameter is taken into account, whereby new time derivative terms emerge during the time-dependent parametric model reduction process.

In this paper different time-varying model reduction techniques for systems with moving loads are presented and discussed. Firstly, LTV systems are considered and the time-dependent projective reduction framework is briefly explained in Sect. 23.2. Since moving loads represent particular LTV systems with only time-dependent input and/or output matrices, we then introduce simple and straightforward reduction approaches for the resulting special cases in Sect. 23.3. These straightforward methods shift the time-dependency or make use of the special structure of the considered problem to compute *time-independent* projection matrices, and thus avoid the emerging time derivative. These approaches are simple and can be used as a basis for comparison with more complex existing techniques that consider the time variability using time-dependent projection matrices (such as e.g. LTV-balanced truncation). In the second part of the paper, we focus on LPV systems and present an advanced time-dependent parametric model reduction approach by matrix interpolation [7, 8] in Sect. 23.4, which makes use of parameter-varying projection matrices $\mathbf{V}(\mathbf{p}(t))$ and $\mathbf{W}(\mathbf{p}(t))$, and takes the emerging time derivatives into account. Some numerical results for the reduction of systems with moving loads applying the proposed methods are reported, compared and discussed in Sect. 23.5. Finally, the conclusions of the contribution and an outlook are given in Sect. 23.6.

23.2 Linear Time-Varying Model Order Reduction

In the following we first consider a high-dimensional linear time-varying system of the form

$$\begin{aligned} \mathbf{E}(t)\,\dot{\mathbf{x}}(t) &= \mathbf{A}(t)\,\mathbf{x}(t) + \mathbf{B}(t)\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}(t)\,\mathbf{x}(t), \end{aligned} \tag{23.1}$$

where $\mathbf{E}(t)$, $\mathbf{A}(t) \in \mathbb{R}^{n \times n}$, $\mathbf{B}(t) \in \mathbb{R}^{n \times m}$ and $\mathbf{C}(t) \in \mathbb{R}^{q \times n}$ are the time-dependent system matrices, $\mathbf{x}(t) \in \mathbb{R}^n$ is the state vector and $\mathbf{u}(t) \in \mathbb{R}^m$, $\mathbf{y}(t) \in \mathbb{R}^q$ represent the inputs and outputs of the system, respectively. The system matrix $\mathbf{E}(t)$ is assumed to be nonsingular for all $t \in [0, T]$. Note that it is also possible to consider second-order systems and reformulate them into the first-order form (23.1).

23.2.1 Time-Dependent Projective Reduction Framework

In projective model order reduction, we aim to find a reduced-order model by approximating the state vector $\mathbf{x}(t)$ on a subspace of lower dimension $r \ll n$. In the time-varying case, the state vector $\mathbf{x}(t)$ might be projected onto a *varying* subspace spanned by the columns of a *time-dependent* projection matrix $\mathbf{V}(t) \in \mathbb{R}^{n \times r}$ [20, 22]. Therefore, the approximation equations read

$$\begin{aligned} \mathbf{x}(t) &\approx \mathbf{V}(t)\,\mathbf{x}_r(t), \\ \dot{\mathbf{x}}(t) &\approx \dot{\mathbf{V}}(t)\,\mathbf{x}_r(t) + \mathbf{V}(t)\,\dot{\mathbf{x}}_r(t), \end{aligned} \tag{23.2}$$

whereby the product rule must be considered in this case for the time derivative of the state vector. Plugging first these both equations into (23.1), and applying thereon a properly chosen time-dependent projection matrix $\mathbf{W}(t)$ which enforces the Petrov-Galerkin condition leads to the time-varying reduced-order model

$$\begin{aligned}\overbrace{\mathbf{W}(t)^T\mathbf{E}(t)\mathbf{V}(t)}^{\mathbf{E}_r(t)}\,\dot{\mathbf{x}}_r(t) &= \left(\overbrace{\mathbf{W}(t)^T\mathbf{A}(t)\mathbf{V}(t)}^{\mathbf{A}_r(t)} - \mathbf{W}(t)^T\mathbf{E}(t)\dot{\mathbf{V}}(t)\right)\mathbf{x}_r(t) + \overbrace{\mathbf{W}(t)^T\mathbf{B}(t)}^{\mathbf{B}_r(t)}\,\mathbf{u}(t),\\ \mathbf{y}_r(t) &= \underbrace{\mathbf{C}(t)\mathbf{V}(t)}_{\mathbf{C}_r(t)}\,\mathbf{x}_r(t).\end{aligned} \tag{23.3}$$

It is noteworthy to mention that the system matrix of the reduced-order model (23.3) not only comprises the reduced matrix $\mathbf{A}_r(t)$, but also includes a further term which depends on the time derivative $\dot{\mathbf{V}}(t)$ of the time-varying projection matrix. This additional term influences the dynamic behaviour of the reduced-order model and should therefore be taken into account.

The usage of *time-dependent* projection matrices (rather than time-independent bases) for the reduction of LTV systems certainly opens up new degrees of freedom, since the bases are time-dependent and not constant anymore. In other words, the time-varying dynamics of the LTV model are not projected onto a constant subspace, but rather onto a *varying* subspace. The reduced basis $\mathbf{V}(t)$—and consequently also the subspace—*adapts itself* during time, provides a more accurate consideration of the arising time variability than a constant basis $\mathbf{V}$ and should therefore (at least theoretically) offer benefits regarding the approximation quality. For the computation of time-dependent projection matrices, however, standard reduction methods such as balanced truncation cannot be directly applied, but must be adapted instead. Furthermore, the time derivative of $\mathbf{V}(t)$ should be approximated numerically (thus increasing the computational effort) and included in the time integration scheme of the reduced-order model [14].

23.3 Straightforward Reduction Approaches for Particular Linear Time-Varying Systems

In the previous section we have seen that the application of time-dependent projection matrices for the reduction of LTV systems comes along with some difficulties and challenges. For the reduction of particular LTV systems, in which only the input and/or output matrices depend on time, the usage of *time-independent* projection matrices $\mathbf{V}$ and $\mathbf{W}$ might be sufficient. In this section we discuss some special, but technically very relevant, cases for LTV systems and propose straightforward approaches to reduce them.

23.3.1 Case 1: Moving Loads

The first case we want to consider is a high-dimensional LTV system with only time-varying input matrix, and all other matrices being time-independent:

$$\begin{aligned} \mathbf{E}\,\dot{\mathbf{x}}(t) &= \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}(t)\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\,\mathbf{x}(t). \end{aligned} \tag{23.4}$$

The time-dependent input matrix describes the position of the moving forces at time t. In the following we present two straightforward approaches to reduce a system in the form above using time-independent projection matrices.

23.3.1.1 Approach 1: Two-Step Method

The first straightforward reduction method is deducted from the two-step approach presented in [21, 3]. The method consists first on a low-rank approximation of the time-varying input matrix, and consequently on applying standard model reduction techniques to the resulting linear time-invariant system with a modified input:

1. The time-variability of the input matrix is shifted to the input variables through a low-rank approximation of the input matrix by $\mathbf{B}(t) \approx \mathbf{B}\,\tilde{\mathbf{B}}(t)$, where $\mathbf{B} \in \mathbb{R}^{n\times\tilde{m}}$ with $\tilde{m} \ll n$ is a constant matrix and $\tilde{\mathbf{B}}(t) \in \mathbb{R}^{\tilde{m}\times m}$. Introducing a new input $\tilde{\mathbf{u}}(t) = \tilde{\mathbf{B}}(t)\,\mathbf{u}(t)$, the original model (23.4) can be transformed to:

$$\begin{aligned} \mathbf{E}\,\dot{\mathbf{x}}(t) &= \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}\,\overbrace{\tilde{\mathbf{B}}(t)\,\mathbf{u}(t)}^{\tilde{\mathbf{u}}(t)}, \\ \mathbf{y}(t) &= \mathbf{C}\,\mathbf{x}(t). \end{aligned} \tag{23.5}$$

2. The resulting multiple-input multiple-output (MIMO) LTI system $(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C})$ can subsequently be reduced by means of any standard reduction approach for LTI systems (balanced truncation, MIMO rational Krylov or MIMO-IRKA, for instance). The reduced-order model is then given by

$$\begin{aligned} \overbrace{\mathbf{W}^T\mathbf{E}\mathbf{V}}^{\mathbf{E}_r}\,\dot{\mathbf{x}}_r(t) &= \overbrace{\mathbf{W}^T\mathbf{A}\mathbf{V}}^{\mathbf{A}_r}\,\mathbf{x}_r(t) + \overbrace{\mathbf{W}^T\mathbf{B}}^{\mathbf{B}_r}\,\overbrace{\tilde{\mathbf{B}}(t)\,\mathbf{u}(t)}^{\tilde{\mathbf{u}}(t)}, \\ \mathbf{y}_r(t) &= \underbrace{\mathbf{C}\mathbf{V}}_{\mathbf{C}_r}\,\mathbf{x}_r(t), \end{aligned} \tag{23.6}$$

where the reduced time-varying input matrix reads $\mathbf{B}_r(t) = \mathbf{B}_r\,\tilde{\mathbf{B}}(t)$.

For the approximation of the input matrix $\mathbf{B}(t)$, other than [21, 3] we simply take the correct input columns $\mathbf{b}_i(t)$ with the moving load acting at corresponding nodes i of a coarse finite element grid and form the low-rank matrix $\mathbf{B}$ with them, without performing a least squares minimization with the basis functions. Note that the two-step approach only provides satisfactory results, if the number of columns $\tilde{m}$ of the low-rank matrix $\mathbf{B}$ is sufficiently large [21]. Otherwise the overall approximation error in the output (due to the approximation error in the input matrix and the model reduction error) can become inadmissibly large. Note also that this reduction method is limited to systems with a known trajectory of the load before the simulation.

23.3.1.2 Approach 2: One-Sided Reduction with Output Krylov Subspace

The second straightforward method uses Krylov subspaces for the reduction and exploits the fact that the only time-varying element in system (23.4) is the input matrix $\mathbf{B}(t)$. Since an input Krylov subspace would yield a time-varying projection matrix

$$\mathbf{V}(t) := \left[\mathbf{A}_{s_0}^{-1}\mathbf{B}(t)\ \mathbf{A}_{s_0}^{-1}\mathbf{E}\mathbf{A}_{s_0}^{-1}\mathbf{B}(t)\ \ldots\ (\mathbf{A}_{s_0}^{-1}\mathbf{E})^{r-1}\mathbf{A}_{s_0}^{-1}\mathbf{B}(t)\right], \tag{23.7}$$

where $\mathbf{A}_{s_0} = \mathbf{A} - s_0\mathbf{E}$, the idea of this approach is to perform a one-sided reduction with $\mathbf{V} = \mathbf{W}$, where the columns of $\mathbf{W}$ form a basis of the output Krylov subspace:

$$\mathbf{W} := \left[\mathbf{A}_{s_0}^{-T}\mathbf{C}^T\ \mathbf{A}_{s_0}^{-T}\mathbf{E}^T\mathbf{A}_{s_0}^{-T}\mathbf{C}^T\ \ldots\ (\mathbf{A}_{s_0}^{-T}\mathbf{E}^T)^{r-1}\mathbf{A}_{s_0}^{-T}\mathbf{C}^T\right]. \tag{23.8}$$

Thereby, time-independent projection matrices are obtained for computing the reduced-order model

$$\begin{aligned}\overbrace{\mathbf{W}^T\mathbf{E}\mathbf{W}}^{\mathbf{E}_r}\,\dot{\mathbf{x}}_r(t) &= \overbrace{\mathbf{W}^T\mathbf{A}\mathbf{W}}^{\mathbf{A}_r}\,\mathbf{x}_r(t) + \overbrace{\mathbf{W}^T\mathbf{B}(t)}^{\mathbf{B}_r(t)}\,\mathbf{u}(t),\\ \mathbf{y}_r(t) &= \underbrace{\mathbf{C}\mathbf{W}}_{\mathbf{C}_r}\,\mathbf{x}_r(t).\end{aligned} \tag{23.9}$$

Although only the first r Taylor coefficients (so-called *moments*) of the transfer function of the original and the reduced model around the expansion points s_0 match due to the application of a one-sided reduction, we obtain time-independent projection matrices with this approach and can therefore get rid of the time derivative $\dot{\mathbf{V}}(t)$. The motivation for this straightforward approach is thus to exploit the special structure of the considered problem to get time-independent projection matrices.

23.3.2 Case 2: Moving Sensors

Now we consider a LTV system with only time-varying output matrix

$$\begin{aligned} \mathbf{E}\,\dot{\mathbf{x}}(t) &= \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}(t)\,\mathbf{x}(t). \end{aligned} \tag{23.10}$$

The time-dependent output matrix describes the position of the moving sensors at time t. This particular LTV system can easily be reduced in the following ways.

23.3.2.1 Approach 1: Two-Step Method

1. We shift the time-variability of the output matrix to the output variables through a low-rank approximation by $\mathbf{C}(t) \approx \tilde{\mathbf{C}}(t)\,\mathbf{C}$, where $\mathbf{C} \in \mathbb{R}^{\tilde{q}\times n}$ with $\tilde{q} \ll n$ is a constant matrix and $\tilde{\mathbf{C}}(t) \in \mathbb{R}^{q\times\tilde{q}}$. Introducing a new output $\tilde{\mathbf{y}}(t) = \mathbf{C}\,\mathbf{x}(t)$, the original model (23.10) can be transformed to:

$$\begin{aligned} \mathbf{E}\,\dot{\mathbf{x}}(t) &= \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \tilde{\mathbf{C}}(t)\,\underbrace{\mathbf{C}\,\mathbf{x}(t)}_{\tilde{\mathbf{y}}(t)}. \end{aligned} \tag{23.11}$$

2. The resulting system $(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C})$ can subsequently be reduced by means of any appropriate multiple-input multiple-output LTI reduction technique. The calculated time-independent projection matrices lead to the reduced-order model

$$\begin{aligned} \overbrace{\mathbf{W}^T\mathbf{E}\mathbf{V}}^{\mathbf{E}_r}\,\dot{\mathbf{x}}_r(t) &= \overbrace{\mathbf{W}^T\mathbf{A}\mathbf{V}}^{\mathbf{A}_r}\,\mathbf{x}_r(t) + \overbrace{\mathbf{W}^T\mathbf{B}}^{\mathbf{B}_r}\,\mathbf{u}(t), \\ \mathbf{y}_r(t) &= \tilde{\mathbf{C}}(t)\,\underbrace{\mathbf{C}\mathbf{V}}_{\mathbf{C}_r}\,\mathbf{x}_r(t), \end{aligned} \tag{23.12}$$

   with the reduced time-varying output matrix $\mathbf{C}_r(t) = \tilde{\mathbf{C}}(t)\,\mathbf{C}_r$.

The approximation of $\mathbf{C}(t)$ is performed by simply taking the output rows with the moving sensor at the corresponding nodes of a coarse finite element grid. Note that the approximation of the output matrix yields additional errors in the output [21].

23.3.2.2 Approach 2: One-Sided Reduction with Input Krylov Subspace

Since in this case an output Krylov subspace would lead to a time-varying projection matrix

$$\mathbf{W}(t) := \left[\mathbf{A}_{s_0}^{-T}\mathbf{C}(t)^T \; \mathbf{A}_{s_0}^{-T}\mathbf{E}^T\mathbf{A}_{s_0}^{-T}\mathbf{C}(t)^T \; \ldots \; (\mathbf{A}_{s_0}^{-T}\mathbf{E}^T)^{r-1}\mathbf{A}_{s_0}^{-T}\mathbf{C}(t)^T\right] \tag{23.13}$$

due to the time-dependent output matrix $\mathbf{C}(t)$, the idea is now to perform a one-sided reduction with $\mathbf{W} = \mathbf{V}$, where the columns of $\mathbf{V}$ form a basis of the input Krylov subspace:

$$\mathbf{V} := \left[\mathbf{A}_{s_0}^{-1}\mathbf{B} \; \mathbf{A}_{s_0}^{-1}\mathbf{E}\mathbf{A}_{s_0}^{-1}\mathbf{B} \; \ldots \; (\mathbf{A}_{s_0}^{-1}\mathbf{E})^{r-1}\mathbf{A}_{s_0}^{-1}\mathbf{B}\right]. \tag{23.14}$$

The reduced model is then given by:

$$\begin{aligned} \overbrace{\mathbf{V}^T\mathbf{E}\mathbf{V}}^{\mathbf{E}_r} \; \dot{\mathbf{x}}_r(t) &= \overbrace{\mathbf{V}^T\mathbf{A}\mathbf{V}}^{\mathbf{A}_r} \; \mathbf{x}_r(t) + \overbrace{\mathbf{V}^T\mathbf{B}}^{\mathbf{B}_r} \; \mathbf{u}(t), \\ \mathbf{y}_r(t) &= \underbrace{\mathbf{C}(t)\mathbf{V}}_{\mathbf{C}_r(t)} \; \mathbf{x}_r(t). \end{aligned} \tag{23.15}$$

Due to the application of a one-sided reduction, only r moments are matched. Nevertheless, the time derivative is avoided, since $\mathbf{V}$ and $\mathbf{W}$ are time-independent ($\dot{\mathbf{V}} = \mathbf{0}$).

23.3.3 *Case 3: Moving Loads and Sensors*

Finally, we consider the combined case with time-varying input *and* output matrices

$$\begin{aligned} \mathbf{E}\,\dot{\mathbf{x}}(t) &= \mathbf{A}\,\mathbf{x}(t) + \mathbf{B}(t)\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}(t)\,\mathbf{x}(t). \end{aligned} \tag{23.16}$$

If the sensor position coincides with the location of the load, then $\mathbf{C}(t) = \mathbf{B}(t)^T$.

23.3.3.1 Approach 1: Two-Step Method

In this case, the respective two-step techniques explained before have to be combined properly:

1. The time-variability of $\mathbf{B}(t)$ is shifted to the input variables and the time-dependency of $\mathbf{C}(t)$ to the output variables, thus obtaining a MIMO LTI system.

2. Time-independent projection matrices are then calculated applying an appropriate model order reduction method to the resulting system $(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C})$. The reduced-order model is finally given by

$$\begin{aligned} \overbrace{\mathbf{W}^T\mathbf{E}\mathbf{V}}^{\mathbf{E}_r}\,\dot{\mathbf{x}}_r(t) &= \overbrace{\mathbf{W}^T\mathbf{A}\mathbf{V}}^{\mathbf{A}_r}\,\mathbf{x}_r(t) + \overbrace{\mathbf{W}^T\mathbf{B}}^{\mathbf{B}_r}\,\overbrace{\tilde{\mathbf{B}}(t)\,\mathbf{u}(t)}^{\tilde{\mathbf{u}}(t)}, \\ \mathbf{y}_r(t) &= \tilde{\mathbf{C}}(t)\,\underbrace{\mathbf{C}\mathbf{V}}_{\mathbf{C}_r}\,\mathbf{x}_r(t). \end{aligned} \tag{23.17}$$

23.3.3.2 Approach 2: Reduction with Modal Truncation

Unfortunately, in this case 3 the application of a one-sided reduction with either an input or an output Krylov subspace would yield time-varying projection matrices $\mathbf{V}(t)$ and $\mathbf{W}(t)$ according to (23.7) or (23.13), respectively. A possible alternative to still obtain time-independent projection matrices and thus get rid of the time derivative $\dot{\mathbf{V}}$ is to use modal truncation as reduction approach. This method only uses the time-independent matrices $\mathbf{A}$ and $\mathbf{E}$ for computing dominant eigenvalues (e.g. with smallest magnitude or smallest real part) and eigenvectors, thus yielding time-independent projection matrices for the reduction.

23.4 Time-Varying Parametric Model Order Reduction

After having considered linear time-varying systems and presented some straightforward approaches to reduce special cases arising in moving load and sensor problems, in this section we focus on linear parameter-varying systems of the form

$$\begin{aligned} \mathbf{E}(\mathbf{p}(t))\,\dot{\mathbf{x}}(t) &= \mathbf{A}(\mathbf{p}(t))\,\mathbf{x}(t) + \mathbf{B}(\mathbf{p}(t))\,\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}(\mathbf{p}(t))\,\mathbf{x}(t). \end{aligned} \tag{23.18}$$

Such systems also exhibit a time-varying dynamic behaviour, since the system matrices explicitly depend on parameters $\mathbf{p}(t)$ which vary over time. Note that moving load and sensor problems can be represented as LPV systems with only parameter-varying input and/or output matrices, if the load and sensor location are considered as time-dependent parameters of the system model. Due to the time-dependency of the parameters, in the next subsection we derive a projection-based, time-varying parametric model order reduction approach called *p(t)MOR*, to obtain a reduced-order model of a LPV system [8, 9]. Based on that, we then adapt the pMOR approach by matrix interpolation [16] to the parameter-varying case, whereby new time derivative terms emerge [7, 9]. For the sake of a concise presentation, the time argument t will be omitted in the state, input and output vectors hereafter.

23.4.1 Projective p(t)MOR

Similarly as explained in Sect. 23.2.1, in the case of projection-based time-dependent parametric model order reduction we aim to approximate the state vector $\mathbf{x}$ by $\mathbf{x} \approx \mathbf{V}(\mathbf{p}(t))\,\mathbf{x}_r$ using a *parameter-varying* projection matrix $\mathbf{V}(\mathbf{p}(t))$. Plugging the corresponding approximation equations for $\mathbf{x}$ and its derivative $\dot{\mathbf{x}}$ in (23.18), and applying thereon a properly chosen projection matrix $\mathbf{W}(\mathbf{p}(t))$ that imposes the Petrov-Galerkin condition yields the reduced-order model

$$\begin{aligned}\mathbf{E}_r(\mathbf{p}(t))\,\dot{\mathbf{x}}_r &= \Big(\mathbf{A}_r(\mathbf{p}(t)) - \mathbf{W}(\mathbf{p}(t))^T\mathbf{E}(\mathbf{p}(t))\dot{\mathbf{V}}(\mathbf{p}(t))\Big)\,\mathbf{x}_r + \mathbf{B}_r(\mathbf{p}(t))\,\mathbf{u},\\ \mathbf{y}_r &= \mathbf{C}_r(\mathbf{p}(t))\,\mathbf{x}_r,\end{aligned} \tag{23.19}$$

with the time-dependent parametric reduced matrices

$$\begin{aligned}\mathbf{E}_r(\mathbf{p}(t)) &= \mathbf{W}(\mathbf{p}(t))^T\mathbf{E}(\mathbf{p}(t))\mathbf{V}(\mathbf{p}(t)), & \mathbf{A}_r(\mathbf{p}(t)) &= \mathbf{W}(\mathbf{p}(t))^T\mathbf{A}(\mathbf{p}(t))\mathbf{V}(\mathbf{p}(t)),\\ \mathbf{B}_r(\mathbf{p}(t)) &= \mathbf{W}(\mathbf{p}(t))^T\mathbf{B}(\mathbf{p}(t)), & \mathbf{C}_r(\mathbf{p}(t)) &= \mathbf{C}(\mathbf{p}(t))\mathbf{V}(\mathbf{p}(t)).\end{aligned} \tag{23.20}$$

The reduced model comprises an additional term depending on the time derivative $\dot{\mathbf{V}}(\mathbf{p}(t))$, which has to be considered during the extension of the matrix interpolation method to the parameter-varying case.

23.4.2 p(t)MOR by Matrix Interpolation

The local pMOR technique of matrix interpolation can be applied to efficiently obtain a parametric reduced-order model from the interpolation of reduced matrices precomputed at different grid points in the parameter space. Similarly as in the classic method [16], the LPV system (23.18) is first evaluated and individually reduced at certain parameter samples $\mathbf{p}_i, i = 1, \ldots, k$ with respective projection matrices $\mathbf{V}_i := \mathbf{V}(\mathbf{p}_i)$ and $\mathbf{W}_i := \mathbf{W}(\mathbf{p}_i)$. The reduced state vectors $\mathbf{x}_{r,i}$ of the independently calculated reduced models

$$\begin{aligned}\mathbf{E}_{r,i}\,\dot{\mathbf{x}}_{r,i} &= \Big(\mathbf{A}_{r,i} - \mathbf{W}_i^T\,\mathbf{E}_i\,\dot{\mathbf{V}}(\mathbf{p}(t))\Big)\,\mathbf{x}_{r,i} + \mathbf{B}_{r,i}\,\mathbf{u},\\ \mathbf{y}_{r,i} &= \mathbf{C}_{r,i}\,\mathbf{x}_{r,i}\end{aligned} \tag{23.21}$$

generally lie in different subspaces and have, therefore, different physical meanings. For this reason, the direct interpolation of the reduced matrices is not meaningful, and hence the local reduced models have to be transformed into a common set of coordinates first. This is performed applying state transformations of the form

$$\begin{aligned} \mathbf{x}_{r,i} &= \mathbf{T}_i\, \hat{\mathbf{x}}_{r,i}, \\ \dot{\mathbf{x}}_{r,i} &= \dot{\mathbf{T}}_i\, \hat{\mathbf{x}}_{r,i} + \mathbf{T}_i\, \dot{\hat{\mathbf{x}}}_{r,i}, \end{aligned} \tag{23.22}$$

with regular matrices $\mathbf{T}_i := \mathbf{T}(\mathbf{p}_i)$, whereby the product rule is required again for the differentiation of $\mathbf{x}_{r,i}$. These state transformations serve to adjust the different *right* local bases $\mathbf{V}_i$ to new bases $\hat{\mathbf{V}}_i = \mathbf{V}_i\,\mathbf{T}_i$. In order to adjust the different *left* local bases $\mathbf{W}_i$ by means of $\hat{\mathbf{W}}_i = \mathbf{W}_i\,\mathbf{M}_i$ as well, the reduced models from (23.21) are subsequently multiplied from the left with regular matrices $\mathbf{M}_i^T$. The resulting reduced and transformed systems are thus given by

$$\begin{aligned} \overbrace{\mathbf{M}_i^T \mathbf{E}_{r,i} \mathbf{T}_i}^{\hat{\mathbf{E}}_{r,i}}\, \dot{\hat{\mathbf{x}}}_{r,i} &= \overbrace{\left(\overbrace{\mathbf{M}_i^T \mathbf{A}_{r,i} \mathbf{T}_i}^{\hat{\mathbf{A}}_{r,i}} - \mathbf{M}_i^T \mathbf{W}_i^T \mathbf{E}_i \dot{\mathbf{V}}(\mathbf{p}(t)) \mathbf{T}_i - \mathbf{M}_i^T \mathbf{E}_{r,i} \dot{\mathbf{T}}_i \right)}^{\hat{\mathbf{A}}_{\mathrm{new}\,r,i}} \hat{\mathbf{x}}_{r,i} + \overbrace{\mathbf{M}_i^T \mathbf{B}_{r,i}}^{\hat{\mathbf{B}}_{r,i}}\, \mathbf{u}, \\ \mathbf{y}_{r,i} &= \underbrace{\mathbf{C}_{r,i} \mathbf{T}_i}_{\hat{\mathbf{C}}_{r,i}}\, \hat{\mathbf{x}}_{r,i}. \end{aligned} \tag{23.23}$$

One possible way to calculate the transformation matrices $\mathbf{T}_i$ and $\mathbf{M}_i$ is based on making the state vectors $\hat{\mathbf{x}}_{r,i}$ compatible with respect to a reference subspace spanned by the columns of the orthogonal matrix $\mathbf{R}$. To this end, the matrices are chosen as $\mathbf{T}_i := (\mathbf{R}^T\, \mathbf{V}_i)^{-1}$ and $\mathbf{M}_i := (\mathbf{R}^T\, \mathbf{W}_i)^{-1}$, where the columns of $\mathbf{R}$ correspond to the r most important directions of $\mathbf{V}_{\mathrm{all}} = [\mathbf{V}_1\ \ldots\ \mathbf{V}_k]$ calculated by a Singular Value Decomposition (SVD) [16].

The resulting system matrix $\hat{\mathbf{A}}_{\mathrm{new}\,r,i}$ not only comprises the expected reduced matrix $\hat{\mathbf{A}}_{r,i}$, but also consists of two further terms that depend on $\dot{\mathbf{V}}(\mathbf{p}(t))$ and $\dot{\mathbf{T}}_i$, respectively. The calculation of these time derivatives that are required for the computation of the reduced-order model will be discussed in the next two sections.

After the transformation of the local models and the computation of the new emerging time derivatives, a parameter-varying reduced-order model for a new parameter value $\mathbf{p}(t)$ is obtained in the online phase by a weighted interpolation between the reduced matrices from (23.23) according to

$$\begin{aligned}\tilde{\mathbf{E}}_r(\mathbf{p}(t)) &= \sum\nolimits_{i=1}^{k} \omega_i(\mathbf{p}(t))\hat{\mathbf{E}}_{r,i}, \quad \tilde{\mathbf{A}}_{\text{new}\,r}(\mathbf{p}(t)) = \sum\nolimits_{i=1}^{k} \omega_i(\mathbf{p}(t))\hat{\mathbf{A}}_{\text{new}\,r,i},\\ \tilde{\mathbf{B}}_r(\mathbf{p}(t)) &= \sum\nolimits_{i=1}^{k} \omega_i(\mathbf{p}(t))\hat{\mathbf{B}}_{r,i}, \qquad \tilde{\mathbf{C}}_r(\mathbf{p}(t)) = \sum\nolimits_{i=1}^{k} \omega_i(\mathbf{p}(t))\hat{\mathbf{C}}_{r,i},\end{aligned} \tag{23.24}$$

where $\sum_{i=1}^{k} \omega_i(\mathbf{p}(t)) = 1$. For simplicity, here we use piecewise linear interpolation of the reduced matrices. Higher order interpolation schemes could also be applied.

23.4.2.1 Time Derivative of V

The time derivative of the projection matrix $\mathbf{V}(\mathbf{p}(t))$ can be numerically calculated using a finite difference approximation. Applying the chain rule first and employing a finite difference method thereon, the time derivative is given by:

$$\dot{\mathbf{V}}(\mathbf{p}(t)) = \frac{\partial \mathbf{V}}{\partial \mathbf{p}}\,\dot{\mathbf{p}} = \frac{\overline{\mathbf{V}}_t - \underline{\mathbf{V}}_t}{\overline{\mathbf{p}}_t - \underline{\mathbf{p}}_t}\,\frac{\mathbf{p}_t - \mathbf{p}_{t-1}}{\Delta t}. \tag{23.25}$$

$\overline{\mathbf{p}}_t$ and $\underline{\mathbf{p}}_t$ denote the upper and lower limit of the interval $[\underline{\mathbf{p}}_t, \overline{\mathbf{p}}_t]$, in which the parameter vector $\mathbf{p}_t$ is located at time instant t. The local bases at these parameter sample points are given by $\overline{\mathbf{V}}_t$ and $\underline{\mathbf{V}}_t$, respectively. The partial derivatives $\frac{\partial \mathbf{V}}{\partial \mathbf{p}}$ for each pair of parameter sample points are calculated in the offline phase of the matrix interpolation approach. In the online phase, the current time derivative $\dot{\mathbf{V}}(\mathbf{p}(t))$ is then computed by multiplying the partial derivative of the corresponding parameter interval at time instant t with $\dot{\mathbf{p}}$, which represents the current velocity of the moving load. Figure 23.1 illustrates the aforementioned intervals and the efficient numerical calculation of the time derivative $\dot{\mathbf{V}}(\mathbf{p}(t))$ by a finite difference approximation using only precomputed local bases. Numerical issues that arise when approximating the derivative via finite differences are out of the scope of this paper and are treated e.g. in [14].

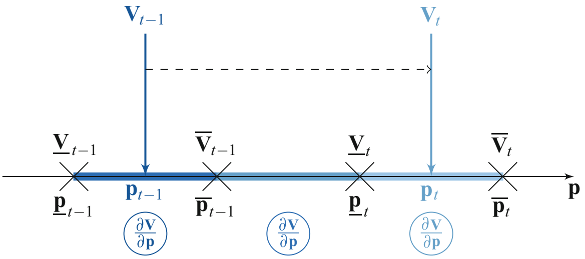

$\mathbf{V}_{t-1}$
$\mathbf{V}_t$
$\underline{\mathbf{V}}_{t-1}$
$\overline{\mathbf{V}}_{t-1}$
$\underline{\mathbf{V}}_t$
$\overline{\mathbf{V}}_t$
$\underline{\mathbf{p}}_{t-1}$
$\mathbf{p}_{t-1}$
$\overline{\mathbf{p}}_{t-1}$
$\underline{\mathbf{p}}_t$
$\mathbf{p}_t$
$\overline{\mathbf{p}}_t$
$\mathbf{p}$
$\frac{\partial \mathbf{V}}{\partial \mathbf{p}}$
$\frac{\partial \mathbf{V}}{\partial \mathbf{p}}$
$\frac{\partial \mathbf{V}}{\partial \mathbf{p}}$

Fig. 23.1 Graphical representation of the calculation of the time derivative $\dot{\mathbf{V}}(\mathbf{p}(t))$ using the local bases $\mathbf{V}_i$ computed at the parameter sample points $\mathbf{p}_i$

23.4.2.2 Time Derivative of T

As explained before, in this paper the transformation matrices $\mathbf{T}_i$ are calculated with $\mathbf{T}_i = (\mathbf{R}^T\mathbf{V}_i)^{-1} := \mathbf{K}^{-1}$. For the computation of the time derivative $\dot{\mathbf{T}}_i$ we make use of the following definition [12, p. 67]:

Definition 1 Let the matrix $\mathbf{K}$ be nonsingular. The time derivative of the inverse matrix is then given by $\frac{\mathrm{d}\mathbf{K}^{-1}}{\mathrm{d}t} = -\mathbf{K}^{-1}\frac{\mathrm{d}\mathbf{K}}{\mathrm{d}t}\mathbf{K}^{-1}$.
This leads to:

$$\dot{\mathbf{T}}_i = \frac{\mathrm{d}\mathbf{K}^{-1}}{\mathrm{d}t} = -(\mathbf{R}^T\mathbf{V}_i)^{-1}\mathbf{R}^T\dot{\mathbf{V}}(\mathbf{p}(t))(\mathbf{R}^T\mathbf{V}_i)^{-1} = -\mathbf{T}_i\mathbf{R}^T\dot{\mathbf{V}}(\mathbf{p}(t))\mathbf{T}_i. \qquad (23.26)$$

23.4.3 p(t)MOR by Matrix Interpolation for Particular Cases

For the reduction of general linear parameter-varying systems the application of time-dependent parametric projection matrices undoubtedly provides an accurate consideration of the arising time variability. Their usage, however, involves some difficulties, like the calculation of the additional derivatives and their incorporation in the numerical simulation of the reduced-order model. Particular LPV systems with only parameter-varying input and/or output matrices, arising e.g. in moving load and sensor problems, can efficiently be reduced using the matrix interpolation approach combined with the usage of *parameter-independent* projection matrices. In the following, this technique is briefly explained for some special cases:

Moving Loads The application of parameter-varying projection matrices $\mathbf{V}(\mathbf{p}(t))$ and $\mathbf{W}(\mathbf{p}(t))$ for the individual reduction of the local systems within matrix interpolation results in a reduced model, where *all* reduced matrices vary with the time-dependent parameter, although the original LPV system only contains variations in the input matrix. In order to get rid of the emerging derivatives and *only* have to interpolate the input matrix in the online phase of matrix interpolation, one-sided reductions with an output Krylov subspace $\mathcal{W} = \mathrm{span}(\mathbf{W})$ should be employed.

Moving Sensors In a similar manner, for the case of a LPV system with only parameter-varying output matrix $\mathbf{C}(\mathbf{p}(t))$ one-sided projections with a single input Krylov subspace $\mathcal{V} = \mathrm{span}(\mathbf{V})$ computed with the input matrix should be performed for the reduction of the sampled models during matrix interpolation. In this way, we obtain parameter-independent projection matrices $\mathbf{V} = \mathbf{W}$ and only have to interpolate the output matrix, thus reducing the computational effort in the online phase.

Moving Loads and Sensors For the combined moving load and sensor example the application of one-sided projections with either input or output Krylov subspaces is not helpful, since both the input and output matrices are parameter-varying in this case. Therefore, parameter-independent projection matrices can only be calculated using modal truncation. By doing so, the reduced-order model only contains parameter variations in the input and output reduced matrices like in the original system.

23.5 Numerical Examples

In this section, we present some numerical results for systems with moving loads.

23.5.1 Timoshenko Beam

The presented reduction approaches are first applied to the finite element model of a simply supported Timoshenko beam of length L subjected to a moving load (Fig. 23.2). Since the moving force $F(t)$ is applied in the negative z-direction and we are only interested in the vertical displacement of the beam, the model described in [15] is adapted from a 3D to a 1D finite element model. Furthermore, both the moving load and/or sensor case are incorporated into the model, yielding time-dependent input and/or output matrices. The resulting single-input single-output second-order system is reformulated into a LTV first-order model of the form

$$\begin{aligned}\overbrace{\begin{bmatrix}\mathbf{F} & \mathbf{0}\\ \mathbf{0} & \mathbf{M}\end{bmatrix}}^{\mathbf{E}}\overbrace{\begin{bmatrix}\dot{\mathbf{z}}\\ \ddot{\mathbf{z}}\end{bmatrix}}^{\dot{\mathbf{x}}}(t) &= \overbrace{\begin{bmatrix}\mathbf{0} & \mathbf{F}\\ -\mathbf{K} & -\mathbf{D}\end{bmatrix}}^{\mathbf{A}}\overbrace{\begin{bmatrix}\mathbf{z}\\ \dot{\mathbf{z}}\end{bmatrix}}^{\mathbf{x}}(t) + \overbrace{\begin{bmatrix}\mathbf{0}\\ \hat{\mathbf{b}}(t)\end{bmatrix}}^{\mathbf{b}(t)}F(t),\\ y(t) &= \underbrace{\begin{bmatrix}\hat{\mathbf{c}}(t)^T & \mathbf{0}^T\end{bmatrix}}_{\mathbf{c}(t)^T}\begin{bmatrix}\mathbf{z}\\ \dot{\mathbf{z}}\end{bmatrix}(t),\end{aligned} \tag{23.27}$$

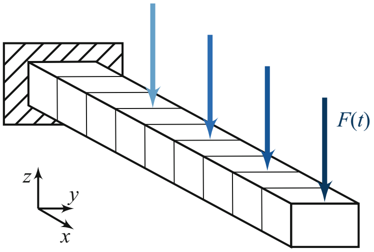


Fig. 23.2 A simply supported Timoshenko beam subjected to a moving force $F(t)$

where the arbitrary nonsingular matrix $\mathbf{F} \in \mathbb{R}^{2N \times 2N}$ is chosen in our case to $\mathbf{F} = \mathbf{K}$ for the aim of stability preservation using a one-sided reduction (cf. [9, 15]). The dimension of the original model is then given by $n = 2 \cdot 2N$ with N finite elements.

Moving Load Case We first consider the reduction of a beam of length $L = 1\,\mathrm{m}$ subjected to a point force moving from the tip to the supporting with a constant velocity v and an amplitude of $F(t) = 20\,\mathrm{N}$. For the numerical simulation we use an implicit Euler scheme with a step size of $dt = 0.001\,\mathrm{s}$. In Fig. 23.3 the simulation results for the different proposed reduction methods are presented. We first apply the standard matrix interpolation (MatrInt) approach using $k = 76$

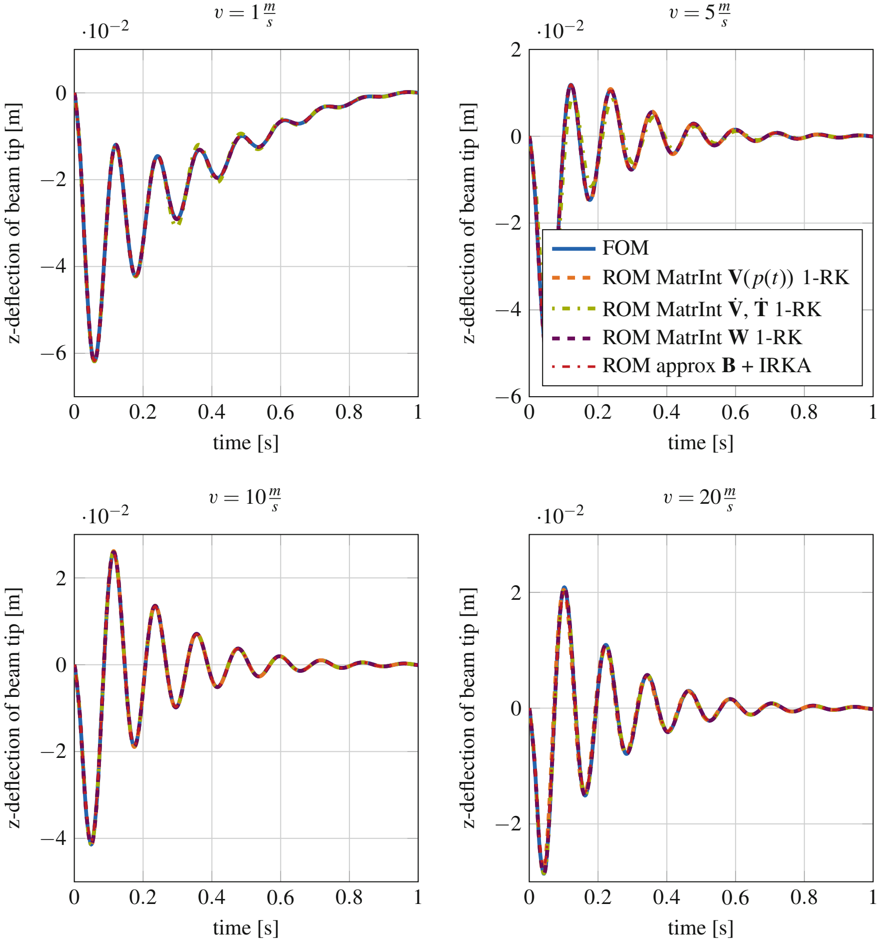


Fig. 23.3 Simulation results for the Timoshenko beam with moving load for different reduction methods and velocities. Original dimension $n = 2 \cdot 2 \cdot 451 = 1804$, reduced dimension $r = 10$. Krylov-based reductions performed with expansion points $s_0 = 0$

Table 23.1 Absolute $\mathscr{L}_2$ output error norms $\|y - y_r\|_{\mathscr{L}_2}$

	MatrInt $\mathbf{V}(p(t))$	MatrInt $\dot{\mathbf{V}}$, $\dot{\mathbf{T}}$	MatrInt $\mathbf{W}$	Approx $\mathbf{B}$ + IRKA
$v = 1\frac{m}{s}$	$4.8e^{-4}$	$1.5e^{-2}$	$3.0e^{-5}$	$1.0e^{-4}$
$v = 5\frac{m}{s}$	$3.0e^{-3}$	$6.8e^{-2}$	$1.8e^{-5}$	$2.0e^{-4}$
$v = 10\frac{m}{s}$	$2.6e^{-3}$	$1.0e^{-3}$	$1.7e^{-5}$	$5.8e^{-5}$
$v = 20\frac{m}{s}$	$2.2e^{-2}$	$5.4e^{-3}$	$1.2e^{-5}$	$3.9e^{-5}$

equidistantly distributed local models with corresponding current input, which are individually reduced applying one-sided projections with input Krylov subspaces ($\mathbf{V}(p(t))$) for $r = 10$. The consideration of the theoretically emerging derivatives $\dot{\mathbf{V}}$ and $\dot{\mathbf{T}}$ according to (23.23) in the matrix interpolation scheme only yields better results than the standard MatrInt method for large velocities of the moving load (as highlighted in Table 23.1). In any case, the application of a single *time-independent* output Krylov subspace ($\mathbf{W}$) during MatrInt and the two-step method ($\tilde{m} = 76$) combined with MIMO-IRKA yields the best results (see Table 23.1).

Moving Load and Sensor Case Now we consider a larger beam of length $L = 50$ m with both moving load and sensor. The observation of the z-deflection of the beam coincides at any time with the position of the moving load, meaning that $\mathbf{c}(p(t))^T = \mathbf{b}(p(t))$. First we apply the matrix interpolation approach and use modal truncation for the individual reduction of the $k = 201$ sampled models constructed with the input and output vectors corresponding to each parameter sample point. Since modal truncation only considers the matrices $\mathbf{A}$ and $\mathbf{E}$ for the reduction and these matrices do not vary over time, we only have to compute *one single pair* of time-independent projection matrices $\mathbf{V}$ and $\mathbf{W}$ in the offline phase. During the online phase, only the parameter-varying input and output vectors have to be interpolated in order to obtain a reduced-order model for each current position of the load/sensor.

Next, we further apply the aforementioned two-step method for the reduction. To this end, the time-varying input and output vectors are first approximated by low-rank matrices $\mathbf{B}$ and $\mathbf{C}$ on a coarse finite element grid. To ensure a proper comparability with MatrInt, we choose the same $\tilde{m} = 201$ nodes where local models were constructed before. The herewith obtained approximated output $y(t)$ and approximation errors are depicted in Fig. 23.4. One can see that the number of chosen columns $\tilde{m}$ is sufficiently large, since the approximation error is adequately small. After that, we both apply two-sided MIMO rational Krylov (2-RK) and MIMO-IRKA for the reduction of the resulting LTI system. Figure 23.4 shows the simulated output for the different explained reduction methods as well as the corresponding absolute and relative $\mathscr{L}_2$ errors. Although all results show a similar behaviour, the matrix interpolation approach combined with modal truncation together with the two-step method by IRKA lead to the smallest errors. Simulations were also conducted with the extended p(t)MOR approach by matrix interpolation considering the time derivatives like in (23.23). Unfortunately, these additional terms make the pencils $(\hat{\mathbf{A}}_{\text{new}\,r,i}, \hat{\mathbf{E}}_{r,i})$ often unstable, yielding unstable interpolated systems and results.

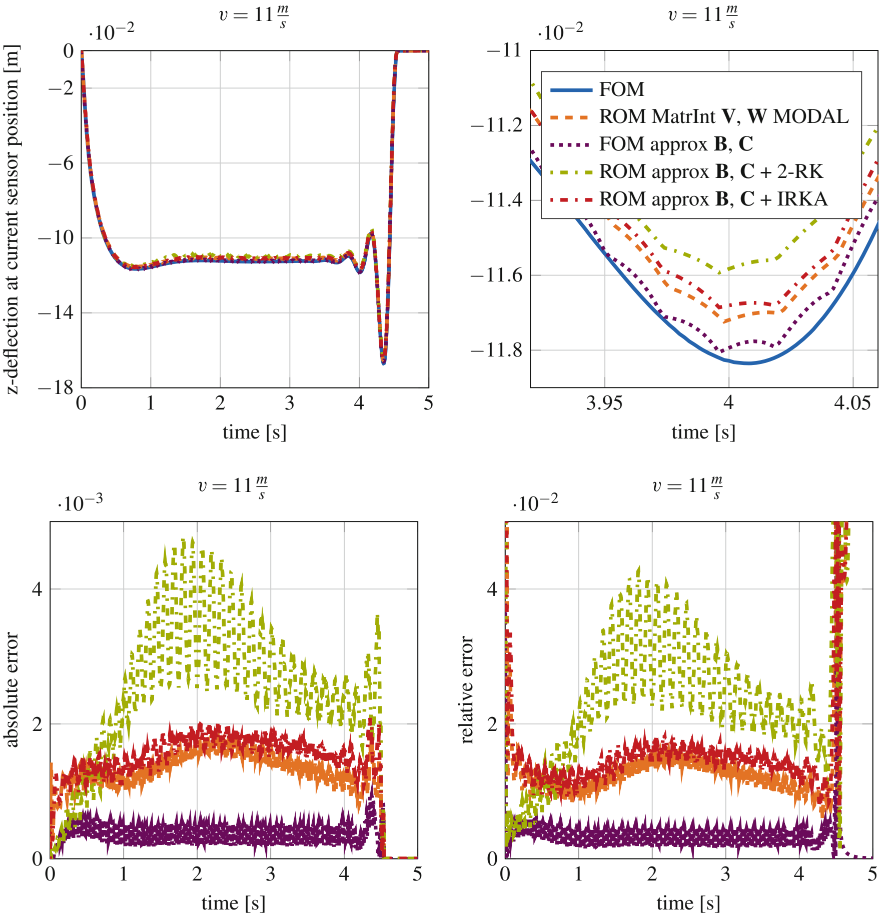


Fig. 23.4 Simulation results for the Timoshenko beam with moving load and sensor for different reduction methods. Original dimension $n = 2 \cdot 2 \cdot 1001 = 4004$, reduced dimension $r = 80$. Krylov-based reductions performed with expansion points $s_0 = 0$

23.5.2 Beam with Moving Heat Source

We now apply the presented techniques on a second example [14], which describes the heat transfer along a beam of length L with a moving heat source. The temperature is observed at the same position as the heat source, thus $\mathbf{c}(t)^T = \mathbf{b}(t)$. In our case, we consider a system dimension of $n = 2500$, apply an input heating source of $u(t) = 50\,°\mathrm{C}$ and use an implicit Euler scheme with a step size of $dt = 1$ s for the time integration. Figure 23.5 shows the simulation results, and the absolute and relative errors for the different employed reduction methods. One interesting observation is that in this case the application of the extended MatrInt

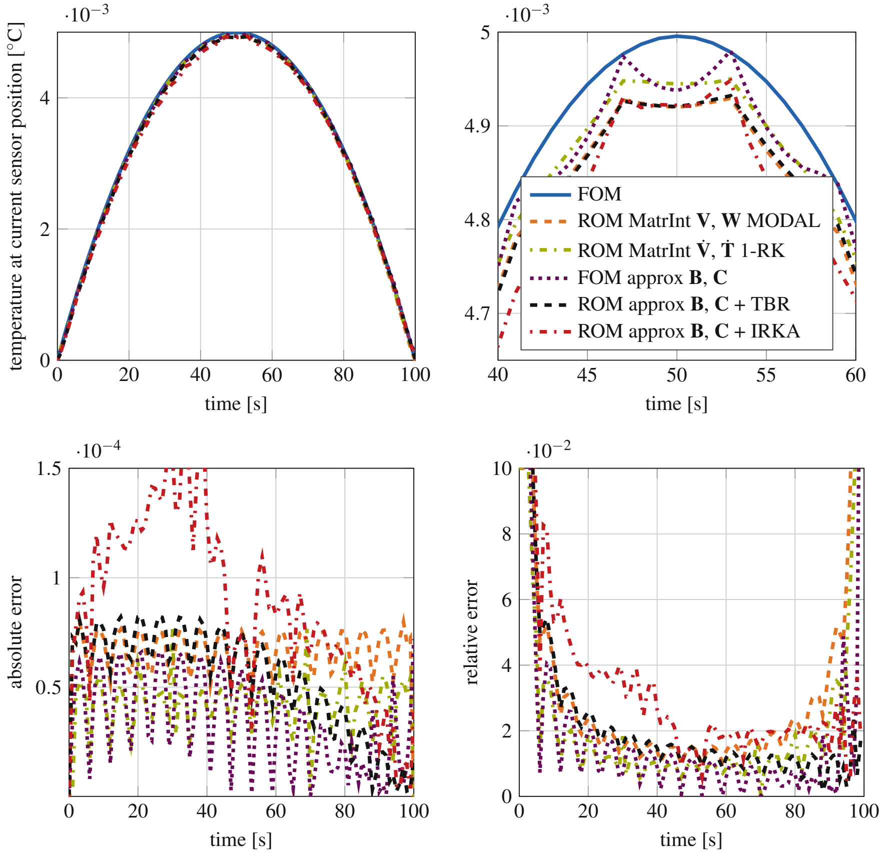


Fig. 23.5 Simulation results for the 1D beam with moving heat source for different reduction methods. Original dimension $n = 2500$, reduced dimension $r = 40$. Krylov-based reductions performed with expansion points $s_0 = 0$

approach with the consideration of the time derivatives yields a slightly better approximation than the classic MatrInt combined with modal truncation ($k = 84$). In general, this fact could also be observed for the previous and some other numerical experiments with higher velocities, as long as the overall interpolated systems were stable. This slightly better approximation can be explained through the more accurate consideration of the arising time variability using time-dependent projection matrices, as opposed to modal truncation which does not consider the moving interactions. The approximation of the time-dependent input and output vectors by low-rank matrices using $\tilde{m} = 84$ nodes, and the subsequent application of balanced truncation (TBR) or MIMO-IRKA for the reduction shows a similar behaviour. Although the extended MatrInt shows in this case the best results, it is difficult to clearly identify a superior method, since all presented approaches are suitable for the reduction of systems with moving loads.

23.6 Conclusions

In this paper, we have presented several time-varying model reduction techniques for systems with moving loads. Such particular, but still frequent problems lead to high-dimensional systems with time-varying input and/or output matrices. For their reduction, time-dependent projection matrices can be applied, thus offering an accurate consideration of the time variation, but leading also to an additional derivative in the reduced model which have to be taken into account. Since moving load problems represent particular LTV systems, we have presented straightforward reduction approaches for some special cases, where time-independent projection matrices are calculated and therefore the emerging time derivative is avoided. Systems with moving loads can also be modeled as special LPV systems, where the input and/or output matrices depend on a time-varying parameter describing the position of the load. In this context we have derived a projection-based, time-varying parametric model reduction approach and extended the matrix interpolation scheme to the parameter-varying case. With the appropriate combination of this method with the application of parameter-independent projection matrices, special LPV systems can be efficiently reduced avoiding the time derivatives. The proposed methods have been tested on two different beam models for both the moving load and/or sensor cases. All techniques have provided similar satisfactory results, showing that all methods are suitable for the reduction of systems with moving loads. In particular, the presented straightforward approaches using time-independent projection matrices are very simple, but may be absolutely sufficient for certain problems. They provide a basis for comparison with more complex techniques that consider the time variability using time-dependent projection matrices. These advanced techniques should be investigated more deeply in the future, especially concerning *general* LTV systems, the increased computational effort due to the time-dependent projection matrices and derivatives, fast-varying load variations and stability preservation in the reduced-order model.

Acknowledgements The authors would like to thank N. Lang, T. Stykel and J. Saak for kindly proving us the 1D heat transfer model used for the numerical example in Sect. 23.5.2. Furthermore, we thank the former and current members of our model order reduction lab (MORLAB) for the fruitful discussions.

References

1. Amsallem, D., Farhat, C.: An online method for interpolating linear parametric reduced-order models. SIAM J. Sci. Comput. **33**(5), 2169–2198 (2011)
2. Antoulas, A.C.: Approximation of Large-Scale Dynamical Systems. SIAM, Philadelphia, PA (2005)
3. Baumann, M., Vasilyev, A., Stykel, T., Eberhard, P.: Comparison of two model order reduction methods for elastic multibody systems with moving loads. Proc. Inst. Mech. Eng. K J. Multibody Dyn. **231**, 48–56 (2016)

4. Baur, U., Beattie, C.A., Benner, P., Gugercin, S.: Interpolatory projection methods for parameterized model reduction. SIAM J. Sci. Comput. **33**(5), 2489–2518 (2011)
5. Baur, U., Benner, P., Feng, L.: Model order reduction for linear and nonlinear systems: a system-theoretic perspective. Arch. Comput. Meth. Eng. **21**(4), 331–358 (2014)
6. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. **57**(4), 483–531 (2015)
7. Cruz Varona, M., Lohmann, B.: Time-varying parametric model order reduction by matrix interpolation. In: Proceedings of the MoRePaS 2015–Model Reduction of Parametrized Systems III (2015)
8. Cruz Varona, M., Geuss, M., Lohmann, B.: p(t)MOR: time-varying parametric model order reduction and applications for moving loads. In: Proceedings of the 8th Vienna Conference on Mathematical Modelling (MATHMOD), vol. 48, pp. 677–678. Elsevier, Amsterdam (2015)
9. Cruz Varona, M., Geuss, M., Lohmann, B.: Zeitvariante parametrische Modellordnungsreduktion am Beispiel von Systemen mit wandernder Last. In: B. Lohmann, G. Roppenecker (Hrsg.) Methoden und Anwendungen der Regelungstechnik – Erlangen-Münchener Workshops 2013 und 2014, pp. 57–70. Shaker-Verlag, Aachen (2015)
10. Fischer, M., Eberhard, P.: Application of parametric model reduction with matrix interpolation for simulation of moving loads in elastic multibody systems. Adv. Comput. Math. **41**(5), 1–24 (2014). https://scholar.google.de/citations?view_op=view_citation&hl=de&user=POrf3BUAAAAJ&sortby=pubdate&citation_for_view=POrf3BUAAAAJ:9yKSN-GCB0IC
11. Fischer, M., Eberhard, P.: Interpolation-based parametric model order reduction for material removal in elastic multibody systems. In: Proceedings ECCOMAS Thematic Conference on Multibody Dynamics (2015)
12. Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th edn. Johns Hopkins University Press, Baltimore (2013)
13. Lang, N., Saak, J., Benner, P.: Model order reduction for systems with moving loads. at-Automatisierungstechnik **62**(7), 512–522 (2014)
14. Lang, N., Saak, J., Stykel, T.: Balanced truncation model reduction for linear time-varying systems. Math. Comput. Model. Dyn. Syst. **22**(4), 267–281 (2016)
15. Panzer, H., Hubele, J., Eid, R., Lohmann, B.: Generating a parametric finite element model of a 3D cantilever Timoshenko beam using MATLAB. TRAC, Lehrstuhl für Regelungstechnik, Technische Universität München (2009)
16. Panzer, H., Mohring, J., Eid, R., Lohmann, B.: Parametric model order reduction by matrix interpolation. at–Automatisierungstechnik **58**(8), 475–484 (2010)
17. Quarteroni, A., Rozza, G.: Reduced Order Methods for Modeling and Computational Reduction, vol. 9. Springer, Berlin (2014)
18. Sandberg, H., Rantzer, A.: Balanced truncation of linear time-varying systems. IEEE Trans. Autom. Control **49**(2), 217–229 (2004)
19. Schilders, W.H.A., van der Vorst, H.A., Rommes, J.: Model Order Reduction: Theory, Research Aspects and Applications. Springer, Berlin (2008)
20. Shokoohi, S., Silverman, L.M., van Dooren, P.M.: Linear time-variable systems: balancing and model reduction. IEEE Trans. Autom. Control **28**(8), 810–822 (1983)
21. Stykel, T., Vasilyev, A.: A two-step model reduction approach for mechanical systems with moving loads. J. Comput. Appl. Math. **297**, 85–97 (2016)
22. Tamarozzi, T., Heirman, G., Desmet, W.: An on-line time dependent parametric model order reduction scheme with focus on dynamic stress recovery. Comput. Methods Appl. Mech. Eng. **268**(0), 336–358 (2014)

Chapter 24
Interpolation Strategy for BT-Based Parametric MOR of Gas Pipeline-Networks

Y. Lu, N. Marheineke, and J. Mohring

Abstract Proceeding from balanced truncation-based parametric reduced order models (BT-pROM) a matrix interpolation strategy is presented that allows the cheap evaluation of reduced order models at new parameter sets. The method extends the framework of model order reduction (MOR) for high-order parameter-dependent linear time invariant systems in descriptor form by Geuss (2013) by treating not only permutations and rotations but also distortions of reduced order basis vectors. The applicability of the interpolation strategy and different variants is shown on BT-pROMs for gas transport in pipeline-networks.

24.1 Introduction

Optimization and control of large transient gas networks require the fast simulation of the underlying parametric partial differential algebraic systems. In this paper we present a surrogate modeling technique that is composed of linearization around stationary states, spatial semi-discretization and model order reduction via balanced truncation (BT). Making use of a matrix interpolation strategy (MIS) in the spirit of [1, 7] we explore its performance for evaluating the BT-pROMs over a wide parameter range of different boundary pressures and temperatures. Our developed variant DTMIS particularly regards possible distortions of the reduced basis vectors.

Y. Lu • N. Marheineke (✉)
FAU Erlangen-Nürnberg, Lehrstuhl Angewandte Mathematik 1, Cauerstr. 11, D-91058 Erlangen, Germany
e-mail: yi.lu@math.fau.de; marheineke@math.fau.de

J. Mohring
Fraunhofer-Institut für Techno- und Wirtschaftsmathematik, Fraunhofer Platz 1, D-67663 Kaiserslautern, Germany
e-mail: jan.mohring@itwm.fraunhofer.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_24

24.2 Modeling Approach for Gas Pipeline-Networks

Proceeding from a nonlinear partial differential algebraic gas network model we perform linearization and spatial semi-discretization to obtain a parametric linear time invariant system as basis for MOR.

Modeling The gas dynamics in a horizontal pipe e can be described by the one-dimensional isothermal Euler equations in terms of pressure p_e and flow rate q_e for space parameter $x \in [x_e^{in}, x_e^{out}]$ and time $t \in [0, t_{end}]$,

$$\partial_t \left(\frac{1}{z} p_e(x,t) \right) + \frac{R_s T}{A_e} \partial_x q_e(x,t) = 0, \tag{24.1a}$$

$$\partial_t q_e(x,t) + A_e \partial_x p_e(x,t) + R_s T \partial_x \left(z \frac{q_e^2(x,t)}{p_e(x,t)} \right) = -\frac{R_s T}{2 A_e D_e} z \lambda \frac{q_e(x,t)|q_e(x,t)|}{p_e(x,t)} \tag{24.1b}$$

with pipe length L_e, diameter D_e, cross-sectional area A_e, temperature T, and specific gas constant R_s. The gas compressibility z and friction λ are empirically given by AGA and Chen formula, respectively, [6], i.e.,

$$z(p_e, T) = 1 + 0.257 \frac{p_e}{p^\star} - 0.533 \frac{p_e T^\star}{p^\star T}$$

with critical pressure $p^\star$ and temperature $T^\star$ values depending on the gas type, and

$$\frac{1}{\sqrt{\lambda(q_e)}} = -2 \log_{10} \left[\frac{\kappa_e}{3.707 D_e} - \frac{5.045}{\text{Re}} \log_{10} \left(\frac{1}{2.826} \left(\frac{\kappa_e}{D_e} \right)^{1.110} + \frac{5.851}{\text{Re}^{0.898}} \right) \right]$$

with Reynolds number $\text{Re}(q_e) = |q_e| D_e / (\eta A_e)$, dynamic gas viscosity η, and pipe roughness κ_e. A network of pipelines can then be modeled as a directed graph $\mathscr{G} = (\mathscr{E}, \mathscr{N})$ where the edges are represented by the pipes $e \in \mathscr{E}$ (with mathematically positive orientation from x_e^{in} to x_e^{out}). The set of nodes $\mathscr{N}$ consists of sources $\mathscr{N}_{\text{in}}$, sinks $\mathscr{N}_{\text{out}}$ and branching (neutral) nodes $\mathscr{N}_{\text{neu}}$. At the branching nodes, mass conservation –known as first Kirchhoff law– and pressure equality in terms of auxiliary variables p are imposed as coupling conditions, i.e.,

$$\sum_{e \in \delta_v^-} q_e(x_e^{out}, t) = \sum_{e \in \delta_v^+} q_e(x_e^{in}, t), \tag{24.1c}$$

$$p_e(x_e^{in}, t) = p(v,t), \; e \in \delta_v^+, \qquad p_e(x_e^{out}, t) = p(v,t), \; e \in \delta_v^-, \qquad v \in \mathscr{N}_{\text{neu}} \tag{24.1d}$$

where δ_v^- and δ_v^+ denote the sets of ingoing and outgoing arcs at $v \in \mathscr{N}_{\text{neu}}$, cf. Fig. 24.1. As boundary conditions we prescribe the pressure profile at the sources

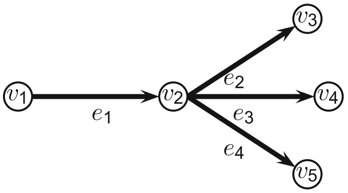


Fig. 24.1 Network topology *Fork*: $\mathscr{G} = (\mathscr{E}, \mathscr{N})$ with source $\mathscr{N}_{\text{in}} = \{v_1\}$, sinks $\mathscr{N}_{\text{out}} = \{v_3, v_4, v_5\}$, as well as $\mathscr{N}_{\text{neu}} = \{v_2\}$, $\delta^-_{v_2} = \{e_1\}$ and $\delta^+_{v_2} = \{e_2, e_3, e_4\}$

and the flow rate at the sinks

$$p(v,t) = f_v(t),\ \ v \in \mathscr{N}_{\text{in}}, \qquad q(v,t) = f_v(t),\ \ v \in \mathscr{N}_{\text{out}}. \tag{24.1e}$$

System (24.1) is supplemented with consistent initial conditions obtained from solving the stationary problem with the boundary conditions (24.1e) evaluated at time $t = 0$

Linearization Expanding around a stationary state $y(x,t) = y^{\text{s}}(x) + \epsilon y^{\text{t}}(x,t) + \mathscr{O}(\epsilon^2)$, $y \in \{p_e, q_e\}$, that is specified by a certain parameter set $\mathbf{p} \in \mathscr{P} \subset \mathbb{R}^d$, the nonlinear system (24.1) decomposes in first order into a stationary subsystem

$$\begin{aligned}
&c^{\text{s}}_{e,1}\,\frac{\mathrm{d}}{\mathrm{d}x}p^{\text{s}}_e + c^{\text{s}}_{e,2}\,\frac{1}{p^{\text{s}}_e} = 0, \quad e \in \mathscr{E}, &&\sum_{e\in\delta^-_v} q^{\text{s}}_e(x^{out}_e) = \sum_{e\in\delta^+_v} q^{\text{s}}_e(x^{in}_e)\\
&p^{\text{s}}_e(x^{in}_e) = p^{\text{s}}(v), \quad e \in \delta^+_v, &&p^{\text{s}}_e(x^{out}_e) = p^{\text{s}}(v),\ \ e \in \delta^-_v, \qquad v \in \mathscr{N}_{\text{neu}}\\
&p^{\text{s}}(v) = f_v(0),\ \ v \in \mathscr{N}_{\text{in}}, &&q^{\text{s}}(v) = f_v(0),\ \ v \in \mathscr{N}_{\text{out}}
\end{aligned}$$

with

$$c^{\text{s}}_{e,1} = 1 - \frac{R_sT}{A_e^2}z^{\text{s}}\left(\frac{q^{\text{s}}_e}{p^{\text{s}}_e}\right)^2, \qquad c^{\text{s}}_{e,2} = \frac{R_sT}{A_e^2}\partial_p z^{\text{s}}(q^{\text{s}}_e)^2 + \frac{R_sT}{2D_eA_e^2}z^{\text{s}}\lambda^{\text{s}}q^{\text{s}}_e|q^{\text{s}}_e|,$$

and a linear transient (correction) subsystem

$$\begin{aligned}
&\partial_t p^{\text{t}}_e + c^{\text{t}}_{e,1}\,\partial_x q^{\text{t}}_e = 0\\
&\partial_t q^{\text{t}}_e + c^{\text{t}}_{e,2}\,\partial_x q^{\text{t}}_e + c^{\text{t}}_{e,3}\,\partial_x p^{\text{t}}_e + c^{\text{t}}_{e,4}\,q^{\text{t}}_e + c^{\text{t}}_{e,5}\,p^{\text{t}}_e = 0, \qquad e \in \mathscr{E}\\
&\sum_{e\in\delta^-_v} q^{\text{t}}_e(x^{out}_e, t) = \sum_{e\in\delta^+_v} q^{\text{t}}_e(x^{in}_e, t)\\
&p^{\text{t}}_e(x^{in}_e, t) = p^{\text{t}}(v,t),\ \ e \in \delta^+_v, \quad p^{\text{t}}_e(x^{out}_e, t) = p^{\text{t}}(v,t),\ \ e \in \delta^-_v, \quad v \in \mathscr{N}_{\text{neu}}\\
&p^{\text{t}}(v,t) = (f_v(t) - f_v(0))/\epsilon,\ \ v \in \mathscr{N}_{\text{in}}, \quad q^{\text{t}}(v,t) = (f_v(t) - f_v(0))/\epsilon,\ \ v \in \mathscr{N}_{\text{out}}
\end{aligned} \tag{24.2}$$

with initial conditions $y^{\mathrm{t}}(x, 0) = 0$, $y \in \{p_e, q_e\}$, and

$$c^{\mathrm{t}}_{e,1} = \frac{R_s T}{A_e}(z^{\mathrm{s}})^2, \qquad c^{\mathrm{t}}_{e,2} = \frac{2R_s T}{A_e} z^{\mathrm{s}} \frac{q^{\mathrm{s}}_e}{p^{\mathrm{s}}_s}, \qquad c^{\mathrm{t}}_{e,3} = A_e - \frac{R_s T}{A_e}\left(\frac{q^{\mathrm{s}}_e}{p^{\mathrm{s}}_s}\right)^2,$$

$$c^{\mathrm{t}}_{e,4} = \frac{R_s T}{A_e}\left(\frac{1}{2D_e} z^{\mathrm{s}} \frac{|q^{\mathrm{s}}_e|}{p^{\mathrm{s}}_e}(2\lambda^{\mathrm{s}} + \partial_q \lambda^{\mathrm{s}} q^{\mathrm{s}}_e) - 2\frac{q^{\mathrm{s}}_e}{(p^{\mathrm{s}}_e)^2}\frac{\mathrm{d}}{\mathrm{d}x} p^{\mathrm{s}}_e\right),$$

$$c^{\mathrm{t}}_{e,5} = \frac{R_s T}{A_e}\left(\frac{1}{2D_e}\lambda^{\mathrm{s}} \frac{|q^{\mathrm{s}}_e|}{p^{\mathrm{s}}_e}\left(\partial_p z^{\mathrm{s}} - z^{\mathrm{s}}\frac{q^{\mathrm{s}}_e}{p^{\mathrm{s}}_e}\right) - 2\left(\frac{q^{\mathrm{s}}_e}{p^{\mathrm{s}}_e}\right)^2 \frac{\mathrm{d}}{\mathrm{d}x} p^{\mathrm{s}}_e \left(\partial_p z^{\mathrm{s}} + \frac{z^{\mathrm{s}}}{p^{\mathrm{s}}_e}\right)\right).$$

The coefficient functions $c^{\mathrm{t}}_{e,j}$, $j = 1, \ldots, 5$, depend not only on the stationary state but also on the model parameters of the pipeline-network and the gas flow. Moreover, note that the flow rate should be regularized before the linearization procedure, i.e., $|q_e| = (q_e^2 + \alpha^2)^{1/2}$, α small, [10].

Semi-discretization As spatial discretization for (24.2) we use a conservative first-order finite-volume-like method on a staggered grid to obtain small discretization stencils. Each pipe is distributed in cells of same length where the pressure p^{t}_e and the mass balance are evaluated at the cell edges and the flow rate q^{t}_e and the momentum balance at the cell midpoints. Sources and sinks are either located on the edges or midpoints of a cell, if pressure or flow rate are given as boundary condition. Neutral nodes are placed at cell boundaries, as suggested in [6]. Function values of p^{t}_e at a midpoint and q^{t}_e at a cell boundary are interpolated. Note that for readability the indices $^{\mathrm{t}}_e$ are suppressed in the stated scheme for a pipe interior,

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} p_{i+1/2} &= -\frac{c_{1,i+1/2}}{\Delta x}(q_{i+1} - q_i) \\ \frac{\mathrm{d}}{\mathrm{d}t} q_i &= -\frac{c_{2,i}}{\Delta x}(q_{i+1/2} - q_{i-1/2}) - \frac{c_{3,i}}{\Delta x}(p_{i+1/2} - p_{i-1/2}) - c_{4,i} q_i - c_{5,i} p_i \end{aligned}$$

with $p_i = (p_{i+1/2} + p_{i-1/2})/2$ and $q_{i+1/2} = (q_{i+1} + q_i)/2$ as well as cell size Δx.

The resulting linear time invariant system (LTIS) of differential algebraic equations (DAE) for the pipeline-network is parameter-dependent, $\Sigma(\mathbf{p})$, $\mathbf{p} \in \mathscr{P} \subset \mathbb{R}^d$,

$$\Sigma(\mathbf{p}): \qquad \mathbf{E}(\mathbf{p})\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = \mathbf{A}(\mathbf{p})\mathbf{x}(t) + \mathbf{B}(\mathbf{p})\mathbf{u}(t), \quad \mathbf{y}(t) = \mathbf{C}(\mathbf{p})\mathbf{x}(t), \qquad (24.3)$$

with system matrices $\mathbf{E}, \mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$ and $\mathbf{C} \in \mathbb{R}^{p \times n}$. The states, inputs and outputs are denoted by $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^p$. The inputs are certainly the boundary conditions, the outputs are taken here as the flow rates at the sources and the pressures at the sinks. As in optimization and control the variation of boundary value profiles (24.1e) is often of interest, we consider a parameter dependence on the boundary pressure $p(v, 0)$, $v \in \mathscr{N}_{\mathrm{in}}$ and the temperature T, i.e., $\mathbf{p} \subset \mathbb{R}^2$. In the following we refer to a sample of N_p different parameter settings and denote the local

LTIS associated with $\mathbf{p}_k$ by Σ_k, $k = 1, \dots, N_p$. It is assumed that Σ_k is stable with the regular pencil $\mathbf{A}_k - \lambda \mathbf{E}_k$. The stability depends, among others, on the applied discretization and is ensured for the discretized gas network under consideration. Moreover, the linearized model is stable as long as the nonlinear model (24.1) is asymptotically stable under "small" perturbations from the stationary state [8]. Note that, whenever possible, we suppress the parameter index $_k$ in the explanations to facilitate the readability.

24.3 BT-MOR for LTIS in Descriptor Form

In the classical method of balanced truncation for ordinary differential equations [4, 15], the original model of order n is first transformed into a balanced form, where the controllability and observability Gramians are diagonally equal. Then, a BT-ROM of order r, $r \ll n$ is obtained by truncating the $(n-r)$ states that are related to the $(n-r)$ smallest Hankel singular values, i.e., diagonal entries of the Gramians.

Considering the full order model (FOM) of DAEs Σ in (24.3), a QZ-decomposition leads to a pencil $\mathbf{A} - \lambda \mathbf{E}$ in the generalized real Schur form. By applying a block-diagonalization [9], Σ can be decoupled into proper and improper subsystems. The spectra of the proper and improper subsystems are the same as the finite and infinite ones of the whole system. Afterwards, the proper and improper subsystems are separately transformed into the balanced form. Whereas the standard BT procedure can be applied to obtain a proper ROM, truncation for the improper subsystem can not be performed in general. If states related to non-zero small HSVs are neglected, the improper ROM may have a finite spectrum with non-negative real parts, which leads to a non-stable inaccurate approximation [11]. In addition, algebraic constraints of the systems might be violated. For example, in case of the gas networks, some of the coupling conditions (Kirchhoff's laws and the pressure equivalence at neutral nodes) may not hold true which implies physically meaningless results. However, states related to zero HSVs can be neglected without affecting the system [14].

Thus, a BT-ROM of order $r = r_f + r_\infty$ is given by

$$\Sigma_r : \qquad \mathbf{E}_r \frac{\mathrm{d}}{\mathrm{d}t} \mathbf{x}_r(t) = \mathbf{A}_r \mathbf{x}_r(t) + \mathbf{B}_r \mathbf{u}(t), \qquad \mathbf{y}_r(t) = \mathbf{C}_r \mathbf{x}_r(t) \tag{24.4a}$$

$$\mathbf{E}_r = \mathbf{W}^{\mathrm{T}} \mathbf{E} \mathbf{V} = \begin{bmatrix} \mathbf{I}_{r_f} & \\ & \mathbf{E}_{r_\infty} \end{bmatrix}, \quad \mathbf{B}_r = \mathbf{W}^{\mathrm{T}} \mathbf{B} = \begin{bmatrix} \mathbf{B}_{r_f} \\ \mathbf{B}_{r_\infty} \end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix} \mathbf{W}_{r_f} & \mathbf{W}_{r_\infty} \end{bmatrix}$$

$$\mathbf{A}_r = \mathbf{W}^{\mathrm{T}} \mathbf{A} \mathbf{V} = \begin{bmatrix} \mathbf{A}_{r_f} & \\ & \mathbf{I}_{r_\infty} \end{bmatrix}, \quad \mathbf{C}_r = \mathbf{C} \mathbf{V} = \begin{bmatrix} \mathbf{C}_{r_f} & \mathbf{C}_{r_\infty} \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} \mathbf{V}_{r_f} & \mathbf{V}_{r_\infty} \end{bmatrix}$$

with its proper and improper subsystems

$$\Sigma_r^{\text{prop}}: \qquad \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}_{r_f}(t) = \mathbf{A}_{r_f}\mathbf{x}_{r_f}(t) + \mathbf{B}_{r_f}\mathbf{u}(t), \qquad \mathbf{y}_{r_f}(t) = \mathbf{C}_{r_f}\mathbf{x}_{r_f}(t) \tag{24.4b}$$

$$\Sigma_r^{\text{improp}}: \quad \mathbf{E}_{r_\infty}\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}_{r_\infty}(t) = \mathbf{x}_{r_\infty}(t) + \mathbf{B}_{r_\infty}\mathbf{u}(t), \qquad \mathbf{y}_{r_\infty}(t) = \mathbf{C}_{r_\infty}\mathbf{x}_{r_\infty}(t). \tag{24.4c}$$

The applied projections **W**, **V** are obviously parameter-dependent, but not orthonormal. They build bases of the (parameter-dependent) rank-r subspaces $\mathscr{W}$, $\mathscr{V}$ in $\mathbb{R}^n$. Analogously, $\mathbf{W}_{r_f}, \mathbf{V}_{r_f}$ and $\mathbf{W}_{r_\infty}, \mathbf{V}_{r_\infty}$ form bases of rank-r_f subspaces $\mathscr{W}_{r_f}, \mathscr{V}_{r_f}$ and rank-r_∞ subspaces $\mathscr{W}_{r_\infty}, \mathscr{V}_{r_\infty}$, respectively. The BT-ROM Σ_r is stable as long as the FOM Σ of (24.3) is stable, [15]. Moreover, since only states related to the improper zero-HSVs are truncated, the DAE-index is preserved, [14].

An error estimate for the system's transfer function **G** in the frequency domain is related to the (decreasingly sorted) proper HSVs σ_i, $i = 1, \ldots, n_f$, [14],

$$\|\mathbf{G} - \mathbf{G}_r\|_{\mathbb{H}_\infty} = \|\mathbf{G}^{\text{prop}} - \mathbf{G}_r^{\text{prop}}\|_{\mathbb{H}_\infty} \leq 2 \sum_{i=r_f+1}^{n_f} \sigma_i$$

with the $\mathbb{H}_\infty$-norm defined as $\|\mathbf{G}\|_{\mathbb{H}_\infty} = \sup_{\omega\in\mathbb{R}} \|\mathbf{G}(i\omega)\|_2$. Hereby, $\mathbf{G}^{\text{prop}}$ and $\mathbf{G}_r^{\text{prop}}$ denote the strictly proper part of **G** and $\mathbf{G}_r$, respectively. Due to the Paley-Wiener Theorem, this error estimate also holds in the time domain [4] where the $\mathbb{H}_\infty$-norm is regarded as the 2-induced operator norm,

$$\|\mathbf{y}(t) - \mathbf{y}_r(t)\|_2 \leq \|\mathbf{G} - \mathbf{G}_r\|_{\mathbb{H}_\infty} \|\mathbf{u}(t)\|_2.$$

24.4 Interpolation for BT-pROMs

BT-MOR is not suitable for online parameter variations in our application because the computational effort (complexity, memory storage) is so extremely high for a large-scale LTIS. Therefore, we suggest an interpolation strategy. Once BT-pROMs $\Sigma_{r,k}$ are computed for different parameter settings $k = 1, \ldots, N_p$, a reduced order model at a new parameter **p** can be efficiently approached by means of interpolation, e.g., by interpolating the transfer functions, the projection spaces or the whole solution (reduced basis method). In [2, 3] an interpolation of BT-pROMs based on the transfer function is investigated and applied to microelectromechanical systems, the approach yields a reduced order model whose size increases with the number of interpolants. In this paper, we explore a (size-preserving) matrix interpolation in the spirit of [1, 7].

Consider the intuitive ansatz

$$\Sigma_r(\mathbf{p}): \quad \mathbf{E}_r(\mathbf{p})\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}_r(t) = \mathbf{A}_r(\mathbf{p})\mathbf{x}_r(t) + \mathbf{B}_r(\mathbf{p})\mathbf{u}(t), \quad \mathbf{y}_r(t) = \mathbf{C}_r(\mathbf{p})\mathbf{x}_r(t) \tag{24.5a}$$

$$\mathbf{M}_r(\mathbf{p}) = \sum_{k=1}^{N_p} \alpha_k(\mathbf{p})\,\mathbf{M}_{r,k}, \qquad \mathbf{M} \in \{\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}\} \tag{24.5b}$$

where the weighting functions α_k are determined by the selected interpolation method. Note that in BT, the states of Σ_k are recombined during the decoupling phase in order to be separated with respect to the finite and infinite spectra. In the MOR phase the states are again recombined such that they can be rearranged according to the HSVs in decreasing order. States related to the small proper HSVs and to the zero-valued improper HSVs are truncated until the local reduced systems $\Sigma_{r,k}$ have the same order r. Thus, the projections $\mathbf{W}_k$, $\mathbf{V}_k$ usually span different rank-r subspaces $\mathscr{W}_k$, $\mathscr{V}_k$ in $\mathbb{R}^n$. Consequently, the reduced states $\mathbf{x}_{r,k}$ have in general no common physical interpretation, which implies that such a interpolation of type (24.5) might not be meaningful.

Generalized Rank-*r* Subspace and Respective Transformation To make sense of the interpolation, all local reduced states $\mathbf{x}_{r,k}$ are transformed in a generalized rank-r subspace $\bar{\mathscr{V}}$. Choosing its basis $\bar{\mathbf{V}}$ requires in general a priori knowledge about the dynamics of the local ROMs. Different strategies are discussed in literature. For example, one of the local bases might act as generalized basis $\bar{\mathbf{V}} = \mathbf{V}_{k_0}$, $k_0 \in \{1, \ldots, N_p\}$, [1]. This is suitable, if all local reduced states lie in the same subspace. In case that the local bases are very different, the generalized basis must catch the most important characteristics of all local ROMs. For this purpose, a Proper Orthogonal Decomposition (POD) [13] can be employed [12], i.e.,

$$\left[\bar{\mathbf{V}}\ \mathbf{U}\right] \Sigma \left[\bar{\mathbf{V}}\ \mathbf{U}\right]^T = \left[\mathbf{V}_1 \cdots \mathbf{V}_{N_p}\right]\left[\mathbf{V}_1 \cdots \mathbf{V}_{N_p}\right]^T.$$

The state transformation $\mathbf{T}_{V,k}$ maps $\mathbf{x}_{r,k}$ in $\bar{\mathscr{V}}$, i.e., $\bar{\mathbf{x}}_r = \mathbf{T}_{V,k}\mathbf{x}_{r,k}$. A transformation proposed in [12]

$$\mathbf{T}_{V,k} = (\bar{\mathbf{V}}^T\mathbf{V}_k)^{-1} \tag{24.6}$$

describes permutations, rotations and length distortions of the basis vectors. Furthermore, it maximizes correlations between each i-th base vectors, $i = 1, \ldots, r$, but minimizes correlations between the i-th and i'-th base vectors, $i \neq i'$ in $\bar{\mathbf{V}}$ and $\mathbf{V}_k$, where the correlations are defined according to the Modal Assurance Criterion

(MAC) [1]

$$MAC(\bar{\mathbf{V}}_{i'}, \mathbf{V}_{k,i}) = \frac{|\langle \bar{\mathbf{V}}_{i'}, \mathbf{V}_{k,i} \rangle|^2}{\langle \bar{\mathbf{V}}_{i'}, \bar{\mathbf{V}}_{i'} \rangle \langle \mathbf{V}_{k,i}, \mathbf{V}_{k,i} \rangle}.$$

However, $\mathbf{T}_{V,k}$ of (24.6) can be singular, if $\bar{\mathbf{V}}$ is orthogonal to $\mathbf{V}_k$. To avoid this crucial weakness, one seeks state transformations such that the sum of the correlations of all i-th base vectors in $\bar{\mathbf{V}}$ and $\mathbf{V}_k$ is maximized.

Theorem 1 *[1, Proposition 4.1.] The optimization problem wrt. the Frobenius norm* $\|\cdot\|_{\mathrm{F}}$

$$\min_{\mathbf{R}_{V,k} \in \mathrm{O}(r)} \left\| \bar{\mathbf{V}} - \mathbf{V}_k \mathrm{R}_{V,k} \right\|_{\mathrm{F}}^2$$

has the unique solution $\mathbf{R}_{V,k} = \mathbf{U}\mathbf{V}^T$*, where* $\mathbf{U}$ *and* $\mathbf{V}$ *are the left and right singular vectors of* $\mathbf{V}_k \bar{\mathbf{V}}^T = \mathbf{U}\Sigma\mathbf{V}^T$.

Proof The first optimality condition together with the uniqueness of the singular value decomposition yields the result. □

The orthogonal mapping $\mathbf{R}_{V,k}$ can handle permutations and rotations of the basis vectors in $\mathbf{V}_k$ wrt. $\bar{\mathbf{V}}$, but cannot capture length distortions. To deal also with distortions, we propose a modification on top of the transformation $\mathbf{R}_{V,k}$.

Theorem 2 *Let* $\bar{\mathbf{V}}_{i'}$ *and* $\tilde{\mathbf{V}}_{k,i}$ *be the i-th column vectors in* $\bar{\mathbf{V}}$ *and* $\bar{\mathbf{V}}_k = \mathbf{V}_k \mathrm{R}_{V,k}$. *The optimization problem*

$$\min_{\gamma_{V,i} \geq 0} \left\| \bar{\mathbf{V}}_{i'} - \gamma_{V,i} \tilde{\mathbf{V}}_{k,i} \right\|_{\mathrm{F}}^2$$

has the unique solution

$$\gamma_{V,i} = \frac{\langle \bar{\mathbf{V}}_{i'}, \tilde{\mathbf{V}}_{k,i} \rangle}{\langle \tilde{\mathbf{V}}_{k,i}, \tilde{\mathbf{V}}_{k,i} \rangle}.$$

Proof The statement follows from the first optimality condition using the fact that $\left\| \bar{\mathbf{V}}_{i'} - \gamma_{V,i} \tilde{\mathbf{V}}_{k,i} \right\|_{\mathrm{F}}^2 = \left\| \bar{\mathbf{V}}_{i'} - \gamma_{V,i} \tilde{\mathbf{V}}_{k,i} \right\|_2^2$. The sign $\gamma_{V,i} \geq 0$ can be particularly concluded from Theorem 1. □

Combining Theorem 1 and Theorem 2 we consider the state transformation

$$\mathbf{T}_{V,k} = \mathbf{R}_{V,k} \mathbf{D}_{V,k}, \qquad \mathbf{D}_{V,k} = \mathrm{diag}(\gamma_{V,1}, \ldots, \gamma_{V,r}). \tag{24.7}$$

Analogously to $\bar{\mathscr{V}}$, $\bar{\mathbf{V}}$ and $\mathbf{T}_{V,k}$, we construct $\bar{\mathscr{W}}$, $\bar{\mathbf{W}}$ and $\mathbf{T}_{W,k}$. This step is necessary, since the local left projections $\mathbf{W}_k$ contain the local left Hankel singular vectors related to the local-preserved HSVs in MOR. Similarly to (24.7), we obtain $\mathbf{T}_{W,k} = \mathbf{R}_{W,k} \mathbf{D}_{W,k}$ with $\mathbf{D}_{W,k} = \mathrm{diag}(\gamma_{W,1}, \ldots, \gamma_{W,r})$. Consequently, the BT-ROM associated with the parameter $\mathbf{p}_k$, $k = 1, \ldots, N_p$, is given with respect to the generalized rank-r subspaces by

$$\hat{\Sigma}_{r,k}: \qquad \hat{\mathbf{E}}_{r,k}\frac{\mathrm{d}}{\mathrm{d}t}\hat{\mathbf{x}}_{r,k}(t) = \hat{\mathbf{A}}_{r,k}\hat{\mathbf{x}}_{r,k}(t) + \hat{\mathbf{B}}_{r,k}\mathbf{u}(t), \qquad \mathbf{y}_{r,k}(t) = \hat{\mathbf{C}}_{r,k}\hat{\mathbf{x}}_{r,k}(t) \tag{24.8}$$

$$\hat{\mathbf{E}}_{r,k} = \mathbf{T}_{W,k}^T\mathbf{E}_{r,k}\mathbf{T}_{V,k}, \qquad \hat{\mathbf{A}}_{r,k} = \mathbf{T}_{W,k}^T\mathbf{A}_{r,k}\mathbf{T}_{V,k},$$

$$\hat{\mathbf{B}}_{r,k} = \mathbf{T}_{W,k}^T\mathbf{B}_{r,k}, \qquad \hat{\mathbf{C}}_{r,k} = \mathbf{C}_{r,k}\mathbf{T}_{V,k}, \qquad \hat{\mathbf{x}}_{r,k} = \mathbf{T}_{V,k}\mathbf{x}_{r,k}.$$

Note that the basis change has no influence on the input-output properties of the system.

Manifold for Interpolation An accurate matrix interpolation (24.5) requires that the variations of the parameter-dependent matrix entries are covered well by the interpolants. For example, the interpolants capture the critical points (wrt. first and second derivatives) of the functions that describe the behavior of the matrix entries on $\mathbf{p} \in \mathscr{P} \subset \mathbb{R}^d$. This certainly presupposes sufficient smoothness of the parameter dependence. Unfortunately, the requirement is hardly fulfilled by the BT-ROMs (24.8) as interpolants. Hence, it may be advantageous to map the matrices into a space where the dependencies can be approximated as well as possible in order to perform the interpolation there and map then the results back to the original space where the BT-ROMs lie. In an appropriate space the matrix entries might be regarded as smooth functions of the parameter. We particularly apply here the concept of a differential Riemannian manifold $\mathscr{M}$ which implies the existence of a tangent space $\mathscr{T}_{\mathbf{M}}$ for each matrix $\mathbf{M} \in \mathscr{M}$ [1, 5, 7].

Let $\mathbf{M}_k$ denote a matrix associated to the parameter $\mathbf{p}_k$, $k = 1, \ldots, N_p$ with N_p sample size. Consider the manifold $\mathscr{M}$ of regular matrices $\mathbb{R}^{r\times r}$ and $\mathbf{M}_k \in \mathscr{M}$. The lifting of the regular matrices $\mathbf{M}_k$ into the tangent space $\mathscr{T}_{\mathbf{M}_{k_0}}$ at a reference matrix $\mathbf{M}_{k_0} \in \mathscr{M}$ can be achieved by the logarithmic mapping which preserves some matrix properties such as symmetric positive definiteness [1]. For that purpose all $\mathbf{M}_k$ must lie in the neighborhood of $\mathbf{M}_{k_0}$ such that $\mathbf{M}_k\mathbf{M}_{k_0}^{-1}$ has a positive spectrum and hence the logarithm $\ln(\mathbf{M}_k\mathbf{M}_{k_0}^{-1})$ is unique and real-valued [7]. Performing the interpolation of the matrices in the tangent space $\mathscr{T}_{\mathbf{M}_{k_0}}$, the result is transformed back into the original manifold $\mathscr{M}$ by means of the exponential mapping. Thus, the corresponding interpolated matrix at $\mathbf{p}$ is given by

$$\mathbf{M}(\mathbf{p}) = \exp\left(\sum_{k=1}^{N_p}\omega(\mathbf{p})\ln\left(\mathbf{M}_k\mathbf{M}_{k_0}^{-1}\right)\right)\mathbf{M}_{k_0}$$

with weight function ω. In case of singular matrices $\mathbf{M}_k$, we consider the manifold of real matrices. The interpolation can then be performed in the linear space at $\mathbf{M}_{k_0}$

by using the affine mapping [1],

$$\mathbf{M}(\mathbf{p}) = \sum_{k=1}^{N_p} \omega(\mathbf{p})\,(\mathbf{M}_k - \mathbf{M}_{k_0}) + \mathbf{M}_{k_0}.$$

The choice of an appropriate reference matrix $\mathbf{M}_{k_0}$ to build the respective tangent space requires in general a priori knowledge about the dependencies of the local BT-ROMs on the parameter, which is hard to analyze. In [5] a heuristic selection criterion is proposed for the case of regular matrices. It is based on the assumption that the entries of $\mathbf{M}_k$ lifted in the tangent space $\mathscr{T}_{\mathbf{M}_{k_0}}$ depend almost linearly on $\mathbf{p}_k = (p_{k,1}, \ldots, p_{k,d}) \in \mathbb{R}^d$. This means that considering $\boldsymbol{\Gamma}_{k_0,k} = \ln(\mathbf{M}_k\mathbf{M}_{k_0}^{-1})$ the respective (i, j)-th matrix entry is approximated by $\gamma_{k_0,k}^{i,j} \approx \alpha_{k_0,0}^{i,j} + \sum_{\ell=1}^{d} \alpha_{k_0,\ell}^{i,j}\, p_{k,\ell}$ with constant coefficients $\alpha_{k_0,\ell}^{i,j}$. Then, the normalized least-squares residual of the sample is used as indicator of the parameter dependence, and the maximal values over all matrix entries are considered as selection criterion for the reference parameter $\mathbf{p}_{k_0}$,

$$k_0^* = \arg\min_{k_0} \mu_{k_0}, \quad \mu_{k_0} = \max_{i,j} \mu_{k_0}^{i,j}, \quad \mu_{k_0}^{i,j} = \frac{\sqrt{\sum_{k=1}^{N_p}(\alpha_{k_0,0}^{i,j} + \sum_{\ell=1}^{d} \alpha_{k_0,\ell}^{i,j} p_{k,\ell} - \gamma_{k_0,k}^{i,j})^2}}{\max_k \gamma_{k_0,k}^{i,j} - \min_k \gamma_{k_0,k}^{i,j}}.$$

Alternatively, one could consider the normalized least-squares residual in the original manifold

$$\theta = \max_{i,j} \theta^{i,j}, \qquad \theta^{i,j} = \frac{\sqrt{\sum_{k=1}^{N_p}(\alpha_0^{i,j} + \sum_{\ell=1}^{d} \alpha_\ell^{i,j} p_{k,\ell} - m_k^{i,j})^2}}{\max_k m_k^{i,j} - \min_k m_k^{i,j}}$$

where $m_k^{i,j}$ denotes the matrix entries of $\mathbf{M}_k$. Comparing θ and μ_{k_0}, the interpolation is performed in the respective tangent space if $\mu_{k_0} \leq \theta$. The prescribed selection criterion can be straightforward transferred to the case of singular matrices, assuming a linear parameter dependence of the matrix entries lifted in the linear space at $\mathbf{M}_{k_0}$, i.e., $\boldsymbol{\Gamma}_{k_0,k} = \mathbf{M}_k - \mathbf{M}_{k_0}$, and in the manifold.

Note that in our application of the gas network, the system matrices $\hat{\mathbf{A}}_{r,k}$ are regular while $\hat{\mathbf{E}}_{r,k}$, $\hat{\mathbf{B}}_{r,k}$ and $\hat{\mathbf{C}}_{r,k}$ are singular.

Interpolation of Decoupled System The BT-ROM $\hat{\Sigma}_{r,k}$ of (24.8) is in general not decoupled in proper and improper subsystems (cf. (24.4)) any more. Hence, any interpolated reduced order model is also not decoupled, as the matrix interpolation (24.5) preserves the structure of the matrices due to the element-wise performance. If the algebraic subsystem of the FOM Σ is parameter-invariant, then there is no interchange between the proper and improper BT-ROMs wrt. the parameter (involving a decoupled form of $\hat{\Sigma}_r$). Hence, the subsystems can be adjusted and interpolated separately. In case of decoupled $\hat{\Sigma}_r$, only $(r_f^2 + r_\infty^2)$

elementary operations are needed to approximate the matrix pencil $\mathbf{A} - \lambda\mathbf{E}$, instead of $2(r_f + r_\infty)^2$ operation for the coupled system. This is more amenable to real-time applications. Note that in the gas networks under consideration the assumption on the FOM holds true, i.e., the algebraic coupling conditions are parameter-independent.

Theorem 3 *Assume that the algebraic part of the FOM (24.3) is parameter-invariant. Then, the proper and improper systems $\Sigma_{r,k}^{\text{prop}}$ and $\Sigma_{r,k}^{\text{improp}}$ of the BT-ROMs (24.4), $k = 1, \dots, N_p$, can be separately transformed into generalized subspaces $\bar{\mathscr{W}} = \bar{\mathscr{W}}_{r_f} \oplus \bar{\mathscr{W}}_{r_\infty}$ and $\bar{\mathscr{V}} = \bar{\mathscr{V}}_{r_f} \oplus \bar{\mathscr{V}}_{r_\infty}$, which are spanned by $\bar{\mathbf{W}}_{r_f}$, $\bar{\mathbf{W}}_{r_\infty}$ and $\bar{\mathbf{V}}_{r_f}$, $\bar{\mathbf{V}}_{r_\infty}$ respectively. Furthermore, the transformation only requires the mapping of $\Sigma_{r,k}^{\text{prop}}$ and $\Sigma_{r,k}^{\text{improp}}$ into $\bar{\mathscr{V}}$.*

Proof To facilitate the readability we suppress the parameter index $_k$. Assume that $\mathrm{T}_{V_{r_f}}$, $\mathrm{T}_{W_{r_f}}$ are the transformations associated with the proper BT-ROM. According to (24.8), the transformed system $\hat{\Sigma}_r^{\text{prop}}$ is given by

$$
\begin{aligned}
&\hat{\mathbf{E}}_{r_f} \frac{\mathrm{d}}{\mathrm{d}t} \hat{\mathbf{x}}_{r_f}(t) = \hat{\mathbf{A}}_{r_f} \hat{\mathbf{x}}_{r_f}(t) + \hat{\mathbf{B}}_{r_f} \mathbf{u}(t), \qquad \mathbf{y}_{r_f}(t) = \hat{\mathbf{C}}_{r_f} \hat{\mathbf{x}}_{r_f}(t) \\
&\hat{\mathbf{E}}_{r_f} = \mathbf{T}_{W_{r_f}}^T \mathbf{E}_{r_f} \mathbf{T}_{V_{r_f}}, \qquad \hat{\mathbf{A}}_{r_f} = \mathbf{T}_{W_{r_f}}^T \mathbf{A}_{r_f} \mathbf{T}_{V_{r_f}}, \qquad \mathbf{E}_{r_f} = \mathbf{I}_{r_f}, \\
&\hat{\mathbf{B}}_{r_f} = \mathbf{T}_{W_{r_f}}^T \mathbf{B}_{r_f}, \qquad \hat{\mathbf{C}}_{r_f} = \mathbf{C}_{r_f} \mathbf{T}_{V_{r_f}}, \qquad \hat{\mathbf{x}}_{r_f} = \mathbf{T}_{V_{r_f}} \mathbf{x}_{r_f}.
\end{aligned}
$$

Since $\hat{\mathbf{E}}_{r_f}$ is regular,

$$
\frac{\mathrm{d}}{\mathrm{d}t} \hat{\mathbf{x}}_{r_f}(t) = \hat{\mathbf{E}}_{r_f}^{-1} \hat{\mathbf{A}}_{r_f} \hat{\mathbf{x}}_{r_f}(t) + \hat{\mathbf{E}}_{r_f}^{-1} \hat{\mathbf{B}}_{r_f} \mathbf{u}(t)
$$

leads to

$$
\mathbf{T}_{V_{r_f}} \frac{\mathrm{d}}{\mathrm{d}t} \mathbf{x}_{r_f}(t) = \mathbf{T}_{V_{r_f}}^{-1} \mathbf{A}_{r_f} \mathbf{T}_{V_{r_f}} \mathbf{x}_{r_f}(t) + \mathbf{T}_{V_{r_f}}^{-1} \mathbf{B}_{r_f} \mathbf{u}(t).
$$

The same can be shown for the improper BT-ROM in an analogue manner. □

24.5 Results and Discussion

Proceeding from a sample of BT-pROMs for gas pipeline-networks we demonstrate the applicability of the matrix interpolation strategy (MIS) for an efficient model order reduction. In particular, we compare different variants (with and without distortion treatment, with original and tangent manifold), regarding the outputs of the interpolated systems.

As test scenario we consider exemplarily the network topology *Fork* visualized in Fig. 24.1 over the time horizon $[0, t_{end}]$, $t_{end} = 48$ [h]. Although it is a rather small network consisting only of four pipes, the results are representative for the application. The pipes $e_1, \ldots, e_4$ have different lengths $L_{e_{1,\ldots,4}} = (16, 45, 7, 38)$ [km], but same diameter $D_e = 1$ [m] and roughness parameter $\kappa_e = 5 \cdot 10^{-5}$ [m]. The last enters with the dynamical gas viscosity $\mu = 10^{-5}$ [kg/(ms)] in the Chen formula for the friction λ. The specific gas constant is $R_s = 448$ [J/(kg K)]. The boundary conditions (24.1e) of the gas network given by $p(v, t) = p_0 + 0.5(1.05p_0 - p_0)(1 - \cos \pi t/t_{end})$ [bar] at $v \in \mathscr{N}_{\text{in}}$ and $q(v, t) = 200$ [kg/s] at $v \in \mathscr{N}_{\text{out}}$ act as inputs, whereas the pressure $p(v, t)$ at $v \in \mathscr{N}_{\text{out}}$ and the flow rate $q(v, t)$ at $v \in \mathscr{N}_{\text{in}}$ are considered as outputs for $t \in [0, t_{end}]$. In addition to the boundary pressure $p_0 \in [55, 65]$ [bar], the temperature $T \in [-20, 20]$ [°C] is regarded as parameter of the model problem, i.e., $\mathbf{p} = (p_0, T) \in \mathscr{P} \subset \mathbb{R}^2$. Note that typical values $p^* = 10^6$, $q^* = 10$ and $t^* = 10^2$ are used to scale pressure, flow rate and time so that the equation system is numerically easier to solve. The stationary problem is determined as follows: using the first Kirchhoff law (24.1c) and the boundary condition, the stationary flow rates of the pipeline-network are evaluated. Afterwards, the stationary pressure of each pipe is calculated by solving an initial value problem for $e_1, \ldots, e_4$.

In the following the original FOMs are of order $n = 35$ due to the spatial discretization with grid size $\Delta x_{e_{1,\ldots,4}} = (6.4, 15, 2.33, 12.67)$ [km]. The BT-ROMs are chosen to be of order $r = 15$. The BT-ROMs decouple in proper and improper subsystems since the algebraic constraints of the FOM are parameter-independent. We solve them by means of the MATLAB routine ode15s (with the default values). Moreover, we use a cubic MIS. Note that the effective choices of the reduced model order and the interpolation order affect quantitatively, but not qualitatively the observed results. Quantitative improvement might be obtained by adapted more sophisticated choices, but this goes beyond the topic of this paper. Focusing on the matrix interpolation we explore here two exemplary model cases that show different parameter-dependent characteristics:

Case I: p_0 sampled at $\{55, 59.5, 65\}$ and $T = 0$,
the generalized bases for the proper and improper BT-ROMs are constructed by using the POD method

Case II: T sampled at $\{-20, -0.49, 20\}$ and $p_0 = 57.7$,
the local bases at $T = -0.49$ are chosen as generalized bases for the proper and improper BT-ROMs

We apply four different MIS variants: DMIS and DTMIS operate without and with distortion treatment on the original manifold, DMMIS and DTDMMIS operate without and with distortion treatment on the tangent manifold.

The approximation quality of the interpolation methods is presented in terms of the relative $\mathscr{L}^2(0, t_{end})$-error in Fig. 24.2, comparing the output of the interpolated system with that of the directly computed BT-ROM. In both model cases our proposed handling of length distortions (Theorem 2) shows a clear improvement. The approximation results are better than the ones achieved with the hitherto

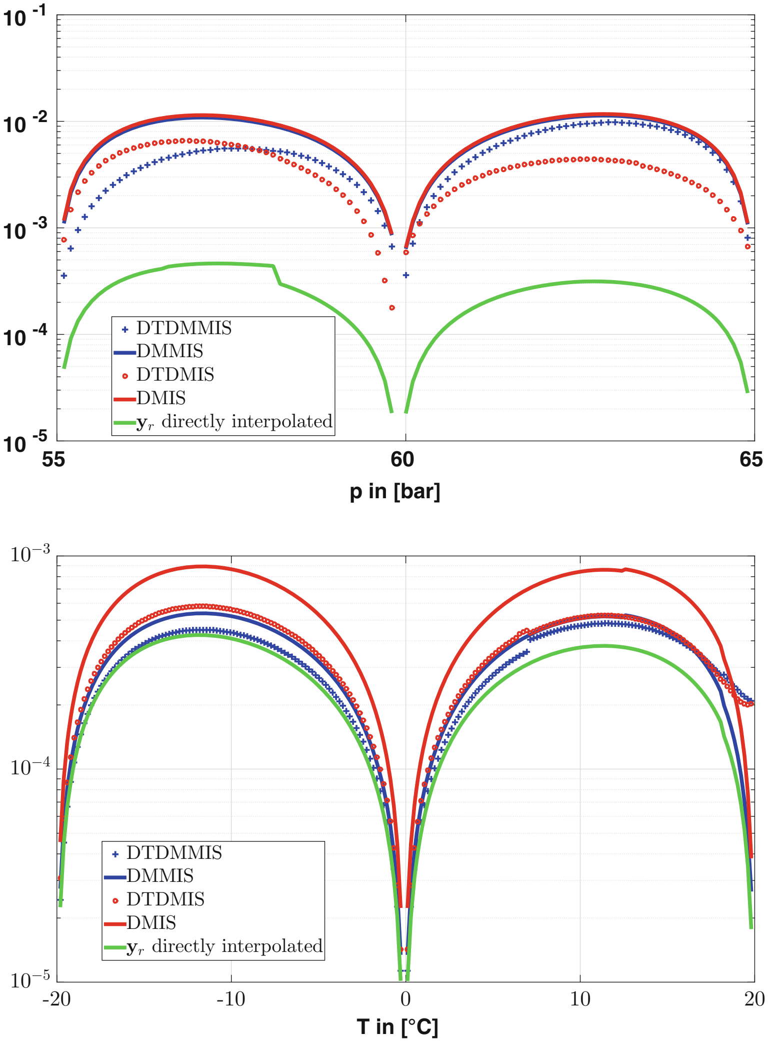


Fig. 24.2 Comparison of different matrix interpolation strategies, relative $\mathscr{L}^2(0, t_{end})$-error between the outputs of the interpolated system and the computed BT-ROM. *Top*: Case I (pressure variations); *bottom*: Case II (temperature variations)

existing matrix interpolation strategy by [7]. The influence of the chosen manifold on the results depends on the considered case. Whereas the use of the original manifold seems beneficially in Case I, it is the tangent manifold in Case II. In total, the results concerning Case II are in size an order better than those of Case I which might be explained by less differences in the underlying local rank-r subspaces. The larger the differences of the local rank-r subspaces, the more difficult is the construction of a generalized subspace (to cover the most important dynamics of the system). In our application, the interpolation results are very robust for temperature variations. Changes in pressure, in contrast, might cause instabilities in the interpolated reduced order models, although the underlying sample of ROMs (interpolants) is stable. This happens for example outside the considered interval $[55, 65]$ in Case I. Developing interpolation techniques that preserve stability is hence topic of recent research. Considering the performance, the combination of MOR and an interpolation strategy is superior to computing directly a ROM at a new parameter setting, because the overall computational costs are dominated by the model order reduction technique. The costs due to our additional distortion handling are marginal.

Figure 24.2 shows additionally the results for directly interpolated outputs $\mathbf{y}_{r,k}$. As it is less error-prone, the direct output interpolation is certainly superior to MIS when only the outputs are of interest. However, optimization and control of transient gas networks require the input-output behavior for large input/output variations over a wide range of parameters. For this purpose, knowledge about the system matrices that belong to the different parameter settings is needed to make possible the cheap and fast evaluation of many reduced order models by help of MIS.

24.6 Conclusion

In this paper we proposed an extension of the matrix interpolation strategy by Geuss [7] for parametric MOR, regarding length distortions of the reduced order basis vector. We showed the applicability and especially the improvement of the results for gas transport in pipeline-networks. The combination of MOR and matrix interpolation allows for the efficient computation of parametric reduced order models and makes optimization and control of large transient gas networks possible. Thereby, the underlying model order reduction technique (here balanced truncation) and the interpolation order that are used are replaceable in view of the desired approximation quality. We remark that non-stable interpolated reduced order models might occur, although the sample of interpolants is stable. Thus, the development of stability-preserving interpolation techniques is addressed in future.

Acknowledgements The German CRC TRR 154 *Mathematical Modelling, Simulation and Optimization using the Example of Gas Networks* is acknowledged.

References

1. Amsallem, D., Farhat, C.: An online method for interpolating linear parametric reduced-order models. SIAM J. Sci. Comput. **33**(5), 2169–2198 (2011)
2. Baur, U., Benner, P.: Model reduction for parametric systems using balanced truncation and interpolation. at-Automatisierungstechnik **57**(8), 411–419 (2009)

3. Baur, U., Benner, P., Greiner, A., Korvink, J.G., Lienemann, J., Moosmann, C.: Parameter preserving model order reduction for MEMS applications. Math. Comput. Model. Dyn. Syst. **17**(4), 297–317 (2011)
4. Benner, P., Mehrmann, V., Sorensen, D.C.: Dimension Reduction of Large-Scale Systems. Lecture Notes in Computational Science and Engineering. Springer, Berlin (2005)
5. Degroote, J., Vierendeels, J., Willcox, K.: Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis. Int. J. Numer. Methods Fluids **63**, 207–230 (2010)
6. Domschke, P.: Adjoint-based control of model and discretization errors for gas transport in networked pipelines. PhD thesis, TU Darmstadt (2011)
7. Geuss, M., Panzer, K., Lohmann, B.: On parametric model order reduction by matrix interpolation. In: European Control Conference (2013)
8. Kailath, T.: Linear Systems. Prentice Hall, New Jersey (1980)
9. Kågström, B., Van Dooren, P.: A generalized state-space approach for the additive decomposition of a transfer matrix. J. Numer. Linear Algebra Appl. **1**(2), 165–181 (1992)
10. Kolb, O.: Simulation and optimization of gas and water supply networks. PhD thesis, TU Darmstadt (2011)
11. Liu, W.Q., Sreeram, V.: Model reduction of singular systems. In: Proceedings of the 39th IEEE Conference on Decision and Control (Sydney, Australia), 2373–2378 (2000)
12. Panzer, K., Mohring, J., Eid, R., Lohmann, B.: Parametric model order reduction by matrix interpolation. at-Automatisierungstechnik **58**(8), 475–484 (2010)
13. Rathinam, M., Petzold, L.R.: A new look at proper orthogonal decomposition. SIAM J. Numer. Anal. **41**(5), 1893–1925 (2003)
14. Stykel, T.: Gramian-based model reduction for descriptor systems. Math. Control Signals Syst. **16**(4), 297–319 (2004)
15. Zhou, K., Doyle, K., Clover, J.D.: Robust and Optimal Control. Prentice Hall, Upper Saddle River (1996)

Chapter 25
Energy Stable Model Order Reduction for the Allen-Cahn Equation

Murat Uzunca and Bülent Karasözen

Abstract The Allen-Cahn equation is a gradient system, where the free-energy functional decreases monotonically in time. We develop an energy stable reduced order model (ROM) for a gradient system, which inherits the energy decreasing property of the full order model (FOM). For the space discretization we apply a discontinuous Galerkin (dG) method and for time discretization the energy stable average vector field (AVF) method. We construct ROMs with proper orthogonal decomposition (POD)-greedy adaptive sampling of the snapshots in time and evaluating the nonlinear function with greedy discrete empirical interpolation method (DEIM). The computational efficiency and accuracy of the reduced solutions are demonstrated numerically for the parametrized Allen-Cahn equation with Neumann and periodic boundary conditions.

25.1 Introduction

The Allen-Cahn equation [2]

$$u_t = \epsilon \Delta u - f(u), \quad (x,t) \in \Omega \times (0,T], \tag{25.1}$$

on a bounded region $\Omega \subset \mathbb{R}^d (d = 1,2)$, is a gradient system in the L_2 norm:

$$u_t = -\frac{\delta \mathscr{E}(u)}{\delta u}. \tag{25.2}$$

M. Uzunca
Department of Industrial Engineering, University of Turkish Aeronautical Association, Ankara, Turkey
e-mail: muzunca@thk.edu.tr

B. Karasözen (✉)
Institute of Applied Mathematics and Department of Mathematics, Middle East Technical University, Ankara, Turkey
e-mail: bulent@metu.edu.tr

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_25

Equation (25.2) is characterized by the minimization of the Ginzburg–Landau energy functional

$$\mathscr{E}(u) = \int_{\Omega} \left(\frac{\epsilon}{2} |\nabla u|^2 + F(u) \right) dx,$$

with a potential functional $F(u)$. The main characteristic of a gradient system is the energy decreasing property:

$$\mathscr{E}(u(t_n)) < \mathscr{E}(u(t_m)), \quad \forall t_n > t_m. \tag{25.3}$$

The Allen-Cahn equation (25.1) was originally introduced to describe the phase of a binary mixture. Nowadays it is used as a model for interface problems in material science, fluid dynamics, image analysis, mean curvature flow, and pattern formation. In (25.1) the unknown u denotes the concentration of the one of the mixture. The parameter ϵ is related to the interfacial width, capturing the dominating effect of reaction kinetics and stays for effective diffusivity. The non-linear term $f(u)$ in (25.1) is given by $f(u) = F'(u)$. Depending on the choice of the potential functional $F(u)$, i.e. the non-linear function $f(u)$, different types of gradient systems occur. The most common potential functions for the Allen–Cahn equation are the convex quartic double-well potential [17] and the non-convex logarithmic potential [7], given respectively by:

$$F(u) = (u^2 - 1)^2/4, \tag{25.4a}$$

$$F(u) = (\theta[(1+u)\ln(1+u) + (1-u)\ln(1-u)] - \theta_c u^2)/2, \tag{25.4b}$$

where θ_c in (25.4b) is the transition temperature. For temperature θ close to θ_c, the logarithmic potential is usually approximated by the convex quartic double-well potential. In case of the quartic double-well potential, $f(u) = u^3 - u$ represents the bi-stable nonlinearity. For the logarithmic potential, it takes the form $f(u) = (\theta/2)\ln((1+u)/(1-u)) - \theta_c u$.

The main characteristic of the Allen-Cahn equation (25.1) is the rapid formation of the transient layers and exponentially slow formation of the terminal layers for very small values of ϵ. This is known as metastability phenomena, characterized by the relative flatness of solutions, where the stable or unstable fixed points coalesce or vanish over long time. These make the numerical computation of the Allen-Cahn equation (25.1) challenging for very small values of ϵ. In the literature for discretization of (25.1) in space, the well-known finite-differences, spectral elements [13], continuous finite elements [26] and local discontinuous Galerkin (LDG) method [21] are used. Several energy stable integrators are developed to preserve the energy decreasing property of the Allen-Cahn equation. For small values of the diffusion parameter ϵ, semi-discretization in space leads to stiff systems. Therefore it is important to design efficient and accurate numerical schemes that are energy stable and robust for small ϵ. Because the explicit methods are not suitable for stiff systems, several energy stable implicit-explicit methods based on the convex splitting of the non-linear term are developed [12, 18, 29].

In this work, we use the symmetric interior penalty Galerkin (SIPG) finite elements for space discretization [4, 27] and the energy stable average vector field (AVF) method [9, 22] for time discretization. The SIPG approximation enables to capture the sharp gradients or singularities locally. On the other hand, the AVF method is the only second order implicit energy stable method for a gradient system. Because the computation of the patterns for small values of ϵ is time-consuming, we consider reduced order modeling which inherits the essential dynamics like the energy decreasing property of the Allen-Cahn equation. In the literature, there are only two papers dealing with the reduced order modeling for Allen-Cahn equation. Using finite difference discretization in space and convex splitting in time, an energy stable reduced order model is derived in [31]. In [24], a non-linear POD/Galerkin reduced order model is applied for efficient computation of the metastable states. An early application of POD to the optimal control of phase field models in material sciences dates back to 2001 [34]. It was shown that for the optimal control of two coupled non-linear PDEs, the solutions of the POD reduced model have nearly the same accuracy as the finite element FOM solutions, whereas the computing time is reduced enormously. Here we apply the greedy proper orthogonal decomposition (PODG) method for the parametrized non-linear parabolic PDEs [15, 19], where the reduced basis functions are formed iteratively by a greedy algorithm for the parameter values such that a POD mode from the matrix of projection error for the parameter value with the largest error is captured and used to enlarge the reduced space. In order to reduce the computational complexity of the function evaluation for the non-linear term in the reduced model, the empirical interpolation method (EIM) [6, 20] and discrete empirical interpolation method (DEIM) [11, 35] are used. The greedy DEIM is included in the adaptive sampling algorithm. We use in the PODG sampling algorithm, the residual-based a posteriori error indicator such that the FOMs are solved only for selected parameter values. We see that the DEIM reduced system is conditionally energy stable whereas the fully discrete system is unconditionally stable. The performance of the PODG approach is illustrated for the Allen Cahn equation with quartic and logarithmic potential functions for different parameters. We want to remark that the majority of the model order reduction (MOR) techniques for parametrized PDEs are projection based, which use the state-space description of the models by numerical simulation. There exists equation-free MOR methods using system responses such as measurements [8]. The data-driven MOR method in the Loewner framework [23] was applied to parametrized systems.

The rest of the paper is organized as follows. In Sect. 25.2, fully discretization of the model problem (25.1) is introduced. In Sect. 25.3, we describe the reduced order modeling together with a POD-greedy sampling algorithm. We present in Sect. 25.4 numerical results for ROMs of the parametrized Allen-Cahn equation with the parameters ϵ and θ.

25.2 Fully Discrete System

In this section, we describe the full discretization of the parametrized form of the (2D) Allen-Cahn equation (25.1) using the symmetric interior penalty Galerkin (SIPG) in space and the second order energy stable average vector field (AVF) method in time. For a certain parameter μ, we denote the parameter dependence of a solution $u(x,t)$ by $u(\mu) := u(x,t;\mu)$, where the parameter μ stands for either the diffusivity ϵ, or the temperature θ in case of logarithmic potential. We also denote the parameter dependence of the non-linear function by $f(u;\mu)$. Then, the variational form of the parametrized Allen-Cahn equation is given as:

$$(\partial_t u(\mu), \upsilon)_\Omega + a(\mu; u(\mu), \upsilon) + (f(u(\mu);\mu), \upsilon)_\Omega = 0, \quad \forall \upsilon \in H^1(\Omega). \tag{25.5}$$

The Allen-Cahn equation was considered in the literature under Dirichlet, Neumann and periodic boundary conditions. Here, we give the SIPG discretization for the homogeneous Neumann boundary conditions [5, 27]; dG discretization for periodic boundary conditions is given in [33]. The SIPG semi-discretized system of (25.5) reads as: for a.e. $t \in (0,T]$, find $u_h(\mu)$ in the SIPG finite element space V_h such that

$$(\partial_t u_h(\mu), \upsilon_h)_\Omega + a_h(\mu; u_h(\mu), \upsilon_h) + (f(u_h(\mu);\mu), \upsilon_h)_\Omega = 0, \qquad \forall \upsilon_h \in V_h, \tag{25.6}$$

with the SIPG bilinear form

$$\begin{aligned} a_h(\mu; u, \upsilon) = \sum_{K \in \mathscr{T}_h} \int_K \epsilon \nabla u \cdot \nabla \upsilon - \sum_{E \in E_h^0} \int_E \{\epsilon \nabla u\} [\upsilon] ds \\ - \sum_{E \in E_h^0} \int_E \{\epsilon \nabla \upsilon\} [u] + \sum_{E \in E_h^0} \frac{\sigma \epsilon}{h_E} \int_E [u][\upsilon] ds, \end{aligned} \tag{25.7}$$

on a triangulation $\mathscr{T}_h$ with triangular elements K and interior edges E having measure h_E. In (25.7), σ denotes the penalty parameter which should be sufficiently large to ensure the stability of the SIPG scheme [5, 27]. For easy notation, we omit the explicit dependence of the discrete solution, bilinear form and the non-linear term on the parameter μ. The solution of (25.6) is given by

$$u_h(x,t) = \sum_{i=1}^{n_K} \sum_{j=1}^{n_q} u_j^i(t) \varphi_j^i(x),$$

where $\varphi_j^i(x)$ and $u_j^i(t)$, $i = 1, \dots, n_k, j = 1, \dots, n_q$, are the basis functions of V_h and the unknown coefficients, respectively. The number n_q denotes the local dimension, depending on the order q of the basis functions, on each triangular element and n_K

is the number of triangular elements. The unknown coefficients and basis functions are defined as vectors:

$$\begin{aligned}\boldsymbol{u} &:= \boldsymbol{u}(t) = (u_1^1(t), u_2^1(t), \ldots, u_{n_q}^{n_K}(t))^T := (u_1(t), u_2(t), \ldots, u_{\mathcal{N}}(t))^T,\\ \boldsymbol{\varphi} &:= \boldsymbol{\varphi}(x) = (\varphi_1^1(x), \varphi_2^1(x), \ldots, \varphi_{n_q}^{n_K}(x))^T := (\varphi_1(x), \varphi_2(x), \ldots, \varphi_{\mathcal{N}}(x))^T.\end{aligned}$$

Here $\mathcal{N} = n_K \times n_q$ denotes the dG degrees of freedom (DoFs). Then, the SIPG semi-discretized system (25.6) leads to the full order model (FOM), in form of a semi-linear system of ordinary differential equations (ODEs):

$$\boldsymbol{M}\boldsymbol{u}_t + \boldsymbol{A}\boldsymbol{u} + \boldsymbol{f}(\boldsymbol{u}) = \boldsymbol{0}, \tag{25.8}$$

for the unknown coefficient vector $\boldsymbol{u}(t)$, where $\boldsymbol{M} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ is the mass matrix, $\boldsymbol{A} \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ is the stiffness matrix, and $\boldsymbol{f}(\boldsymbol{u}) \in \mathbb{R}^{\mathcal{N}}$ is the non-linear vector of unknown coefficients $\boldsymbol{u}$, whose i-th entry is given by $\boldsymbol{f}_i(\boldsymbol{u}) = (f(u_h(t)), \varphi_i(x))_\Omega$, $i = 1, \ldots, \mathcal{N}$.

For the temporal discretization, we consider the uniform partition $0 = t_0 < t_1 < \ldots < t_J = T$ of the time interval $[0, T]$ with the uniform time step-size $\Delta t = t_{n+1} - t_n, n = 0, 1, \ldots, J-1$. As the time integrator, we use the AVF method [9, 22] which preserves the energy decreasing property without restriction of the step size Δt. The AVF method for a general gradient system $\dot{y} = -\nabla G(y)$ is given as:

$$y_{n+1} = y_n - \Delta t \int_0^1 \nabla G(\tau y_{n+1} + (1-\tau)y_n)d\tau.$$

The application of the AVF time integrator to (25.8) leads to the fully discrete system

$$\boldsymbol{M}\boldsymbol{u}^{n+1} - \boldsymbol{M}\boldsymbol{u}^n + \frac{\Delta t}{2}\boldsymbol{A}(\boldsymbol{u}^{n+1} + \boldsymbol{u}^n) + \Delta t \int_0^1 \boldsymbol{f}(\tau\boldsymbol{u}^{n+1} + (1-\tau)\boldsymbol{u}^n)d\tau = \boldsymbol{0}. \tag{25.9}$$

25.2.1 Energy Stability of the Full Order Model

Now, we prove that the SIPG-AVF full discretized gradient system is unconditionally energy stable. The SIPG discretized energy function of the continuous energy $\mathscr{E}(u)$ at a time $t_n = n\Delta t$ is given by:

$$\begin{aligned}\mathscr{E}_h(u_h^n) =& \frac{\epsilon}{2}\left\|\nabla u_h^n\right\|_{L^2(\Omega)}^2 + (F(u_h^n), 1)_\Omega\\ &+ \sum_{E\in E_h^0}\left(-(\{\epsilon\nabla u_h^n\}, [u_h^n])_E + \frac{\sigma\epsilon}{2h_E}([u_h^n], [u_h^n])_E\right),\end{aligned} \tag{25.10}$$

where $u_h^n := u_h(t_n) \in V_h$. Applying the AVF time integrator to the semi-discrete system (25.6) and using the bilinearity of a_h, we get for $n = 0, 1, \ldots, J-1$

$$\begin{aligned}&\frac{1}{\Delta t}(u_h^{n+1} - u_h^n, \upsilon_h)_\Omega + \frac{1}{2}a_h(u_h^{n+1} + u_h^n, \upsilon_h)\\&+ \int_0^1 (f(\tau u_h^{n+1} + (1-\tau)u_h^n), \upsilon_h)_\Omega d\tau = 0.\end{aligned}$$

Choosing $\upsilon_h = u_h^{n+1} - u_h^n$ and using the algebraic identity $(a+b)(a-b) = a^2 - b^2$

$$\begin{aligned}&\frac{1}{\Delta t}(u_h^{n+1} - u_h^n, u_h^{n+1} - u_h^n)_\Omega + \frac{1}{2}a_h(u_h^{n+1}, u_h^{n+1}) - \frac{1}{2}a_h(u_h^n, u_h^n)\\&+ \int_\Omega \left[\int_0^1 (f(\tau u_h^{n+1} + (1-\tau)u_h^n)(u_h^{n+1} - u_h^n)d\tau\right] dx = 0.\end{aligned} \tag{25.11}$$

By the change of variable $z_h = \tau u_h^{n+1} + (1-\tau)u_h^n$, we get

$$\int_0^1 (f(\tau u_h^{n+1} + (1-\tau)u_h^n)(u_h^{n+1} - u_h^n)d\tau = \int_{u_h^n}^{u_h^{n+1}} f(z_h)dz = F(u_h^{n+1}) - F(u_h^n). \tag{25.12}$$

Finally, substituting (25.12) into (25.11), using (25.7) and (25.10), we obtain

$$\mathscr{E}_h(u_h^{n+1}) - \mathscr{E}_h(u_h^n) = -\frac{1}{\Delta t}\|u_h^{n+1} - u_h^n\|^2_{L^2(\Omega)} \leq 0,$$

which implies that $\mathscr{E}_h(u_h^{n+1}) \leq \mathscr{E}_h(u_h^n)$ for any time step size $\Delta t > 0$.

25.3 Model Order Reduction for Gradient Systems

In this section, we describe the construction of the reduced order model (ROM) and DEIM of the non-linear term for the SIPG discretization. We also show that the reduced solutions using DEIM provides conditional energy stability of the discrete energy function.

25.3.1 Reduced Order Model

The ROM solution $u_{h,r}(x,t)$ of dimension $N \ll \mathscr{N}$ is formed by approximating the solution $u_h(x,t)$ in a subspace $V_{h,r} \subset V_h$ spanned by a set of L^2-orthogonal basis functions $\{\psi_i\}_{i=1}^N$ of dimension N, and then projecting onto $V_{h,r}$:

$$u_h(x,t) \approx u_{h,r}(x,t) = \sum_{i=1}^{N} u_{i,r}(t)\psi_i(x), \qquad (\psi_i(x), \psi_j(x))_\Omega = \delta_{ij}, \tag{25.13}$$

where $\boldsymbol{u}_r(t) := (u_{1,r}(t), \dots, u_{N,r}(t))^T$ is the coefficient vector of the reduced solution. Then, the SIPG weak formulation for ROM reads as: for a.e. $t \in (0,T]$, find $u_{h,r}(x,t) \in V_{h,r}$ such that

$$(\partial_t u_{h,r}, \upsilon_{h,r})_\Omega + a_h(u_{h,r}, \upsilon_{h,r}) + (f(u_{h,r}), \upsilon_{h,r})_\Omega = 0, \quad \forall \upsilon_{h,r} \in V_{h,r} \tag{25.14}$$

Since the reduced basis functions $\{\psi_i\}_{i=1}^N \subset V_{h,r}$ also belong to the space V_h, they can be expanded by finite element basis functions $\{\varphi_i(x)\}_{i=1}^{\mathcal{N}}$ as:

$$\psi_i(x) = \sum_{j=1}^{\mathcal{N}} \Psi_{j,i}\varphi_j(x), \qquad \Psi_{\cdot,i}^T \boldsymbol{M} \Psi_{\cdot,j} = \delta_{ij}. \tag{25.15}$$

The coefficient vectors of the reduced basis function are collected in the columns of the matrix $\boldsymbol{\Psi} = [\Psi_{\cdot,1}, \dots, \Psi_{\cdot,N}] \in \mathbb{R}^{\mathcal{N}\times N}$. The coefficient vectors of FOM and ROM solutions are related by $\boldsymbol{u} \approx \boldsymbol{\Psi}\boldsymbol{u}_r$. Substituting this relation together with (25.13) and (25.15) into the system (25.14), we obtain for the unknown coefficient vectors the reduced semi-discrete ODE system:

$$\partial_t \boldsymbol{u}_r + \boldsymbol{A}_r \boldsymbol{u}_r + \boldsymbol{f}_r(\boldsymbol{u}_r) = \boldsymbol{0}, \tag{25.16}$$

with the reduced stiffness matrix $\boldsymbol{A}_r = \boldsymbol{\Psi}^T \boldsymbol{A} \boldsymbol{\Psi}$ and the reduced non-linear vector $\boldsymbol{f}_r(\boldsymbol{u}_r) = \boldsymbol{\Psi}^T \boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{u}_r)$. The construction of the reduced basis functions $\{\psi_i\}_{i=1}^N$ is discussed in Sect. 25.3.4.

25.3.2 Discrete Empirical Interpolation Method (DEIM)

Although the dimension of the reduced system (25.16) is small, $N \ll \mathcal{N}$, the computation of the reduced non-linear vector $\boldsymbol{f}_r(\boldsymbol{u}_r) = \boldsymbol{\Psi}^T \boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{u}_r)$ still depends on the dimension $\mathcal{N}$ of the full system. In order to reduce the online computational cost, we apply the DEIM [11] to approximate the non-linear vector $\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{u}_r) \in \mathbb{R}^{\mathcal{N}}$ from a $M \ll \mathcal{N}$ dimensional subspace spanned by non-linear vectors $\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{u}_r(t_n))$, $n = 1, \dots, J$. Let $M \ll \mathcal{N}$ orthonormal basis functions $\{W_i\}_{i=1}^M$ are given. We set the matrix $\boldsymbol{W} := [W_1, \dots, W_M] \in \mathbb{R}^{\mathcal{N}\times M}$ (the functions W_i are computed successively during the greedy iteration in EIM, whereas here, in DEIM, the functions W_i are computed priori by POD and then they are used in the greedy iteration). Then, we can use the approximation $\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{u}_r) \approx \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi}\boldsymbol{u}_r)$, where

$\boldsymbol{f}_m(\boldsymbol{\Psi}\boldsymbol{u}_r) = \boldsymbol{P}^T\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{u}_r) \in \mathbb{R}^M$ and the matrix $\boldsymbol{Q} = \boldsymbol{W}(\boldsymbol{P}^T\boldsymbol{W})^{-1} \in \mathbb{R}^{\mathcal{N}\times M}$ is precomputable. For the details of the computation of the reduced non-linear vectors we refer to the greedy DEIM algorithm [11]. For continuous finite element and finite volume discretizations, the number of flops for the computation of bilinear form and nonlinear term depends on the maximum number of neighbor cells [15]. In the case of dG discretization, due to its local nature, it depends only on the number of nodes in the local cells. For instance, in the case of SIPG with linear elements ($n_q = 3$), for each degree of freedom, integrals have to be computed on a single triangular element [25], whereas in the case of continuous finite elements, integral computations on 6 neighbor cells are needed [3]. Since the AVF method is an implicit time integrator, at each time step, a non-linear system of equations has to be solved by Newton's method. The reduced Jacobian has a diagonal block structure for the SIPG discretization, which is easily invertible [25], and requires $O(n_qM)$ operations with DEIM.

25.3.3 Energy Stability of the Reduced Solution

The energy stability of the DEIM reduced order model is proved in the same way as for the FOM. Applying the AVF time integrator to the semi-discrete system (25.14), choosing $\upsilon_{h,r} = u_{h,r}^{n+1} - u_{h,r}^n$, and using the algebraic identity $(a+b)(a-b) = a^2 - b^2$ and the bilinearity of a_h, we obtain

$$\begin{aligned} &\frac{1}{\Delta t}(u_{h,r}^{n+1} - u_{h,r}^n, u_{h,r}^{n+1} - u_{h,r}^n)_\Omega + \frac{1}{2}a_h(u_{h,r}^{n+1}, u_{h,r}^{n+1}) - \frac{1}{2}a_h(u_{h,r}^n, u_{h,r}^n) \\ &\quad + \int_0^1 \left[\int_\Omega f(\tau u_{h,r}^{n+1} + (1-\tau)u_{h,r}^n)(u_{h,r}^{n+1} - u_{h,r}^n)dx\right] d\tau = 0. \end{aligned} \tag{25.17}$$

Let us set the averaged reduced solution $z_{h,r} = \tau u_{h,r}^{n+1} + (1-\tau)u_{h,r}^n$, and the averaged coefficient vector $\boldsymbol{z}_r = \tau \boldsymbol{u}_r^{n+1} + (1-\tau)\boldsymbol{u}_r^n$ of the reduced system. Then, from the integral term in (25.17):

$$\begin{aligned} \int_0^1 \left[\int_\Omega f(z_{h,r})(u_{h,r}^{n+1} - u_{h,r}^n)dx\right] d\tau &= \int_0^1 \left[\sum_{i=1}^N (u_{i,r}^{n+1}(t) - u_{i,r}^n(t)) \overbrace{\int_\Omega f(z_{h,r})\psi_i(x)dx}^{(\boldsymbol{\Psi}^T\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{z}_r))_i}\right] d\tau \\ &= \int_0^1 (\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n))^T\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{z}_r))d\tau. \end{aligned}$$

On the left hand side of (25.17), using the DEIM approximation $\boldsymbol{f}(\boldsymbol{\Psi}\boldsymbol{z}_r) \approx \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi}\boldsymbol{z}_r)$, adding and subtracting the term $\int_0^1 \left[\int_\Omega f(z_{h,r})(u_{h,r}^{n+1} - u_{h,r}^n)dx\right] d\tau$, and using the integral mean theorem, we obtain

$$\begin{aligned}\mathscr{E}_h(u_{h,r}^{n+1}) - \mathscr{E}_h(u_{h,r}^n) =& -\frac{1}{\Delta t}\|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}^2 \\ &+ \int_0^1 (\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n))^T(\boldsymbol{f}(\boldsymbol{\Psi} z_r) - \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi} z_r))d\tau \\ =& -\frac{1}{\Delta t}\|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}^2 \\ &+ \langle \boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n), \boldsymbol{f}(\boldsymbol{\Psi}\tilde{z}_r^n) - \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi}\tilde{z}_r^n)\rangle,\end{aligned}$$

for some $\tilde{z}_r^n$ between $\boldsymbol{u}_r^n$ and $\boldsymbol{u}_r^{n+1}$, and $\langle \cdot, \cdot \rangle$ denoting the Euclidean inner product. Applying the Cauchy-Schwarz inequality, we get

$$\langle \boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n), \boldsymbol{f}(\boldsymbol{\Psi}\tilde{z}_r^n) - \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi}\tilde{z}_r^n)\rangle \le \|\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)\|_2 \|\boldsymbol{f}(\boldsymbol{\Psi}\tilde{z}_r^n) - \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi}\tilde{z}_r^n)\|_2.$$

Using the a priori error bound [3, 10, 11], we have

$$\|\boldsymbol{f}(\boldsymbol{\Psi}\tilde{z}_r^n) - \boldsymbol{Q}\boldsymbol{f}_m(\boldsymbol{\Psi}\tilde{z}_r^n)\|_2 \le \|(\boldsymbol{P}^T\boldsymbol{W})^{-1}||_2\|(\boldsymbol{I} - \boldsymbol{W}\boldsymbol{W}^T)\boldsymbol{f}(\boldsymbol{\Psi}\tilde{z}_r^n)||_2.$$

Using the equivalent weighted-Euclidean inner product form of the L^2-norm on the reduced space $V_{h,r}$, we have

$$\begin{aligned}\|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}^2 =& (\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)^T\boldsymbol{M}_r(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n) = (\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)^T\boldsymbol{\Psi}^T\boldsymbol{M}\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n) \\ =& (\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)^T\boldsymbol{\Psi}^T\boldsymbol{R}^T\boldsymbol{R}\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n) = \|\boldsymbol{R}\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)\|_2^2,\end{aligned}$$

where $\boldsymbol{M}_r = \boldsymbol{\Psi}^T\boldsymbol{M}\boldsymbol{\Psi}$ is the reduced mass matrix (indeed it is the identity matrix, $\boldsymbol{\Psi}$ is $\boldsymbol{M}$-orthogonal), and $\boldsymbol{R}$ is the Cholesky factor of the mass matrix $\boldsymbol{M}$ (i.e. $\boldsymbol{M} = \boldsymbol{R}^T\boldsymbol{R}$). Thus, we get the identity

$$\begin{aligned}\|\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)\|_2 =& \|\boldsymbol{R}^{-1}\boldsymbol{R}\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)\|_2 \le \|\boldsymbol{R}^{-1}\|_2\|\boldsymbol{R}\boldsymbol{\Psi}(\boldsymbol{u}_r^{n+1} - \boldsymbol{u}_r^n)\|_2 \\ =& \|\boldsymbol{R}^{-1}\|_2\|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}.\end{aligned}$$

Using the above identity, we obtain for the energy difference:

$$\begin{aligned}\mathscr{E}_h(u_{h,r}^{n+1}) - \mathscr{E}_h(u_{h,r}^n) \le& \|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}^2 \times \\ & \left(-\frac{1}{\Delta t} + \frac{\|\boldsymbol{R}^{-1}\|_2\|(\boldsymbol{P}^T\boldsymbol{W})^{-1}||_2\|(\boldsymbol{I} - \boldsymbol{W}\boldsymbol{W}^T)\boldsymbol{f}(\boldsymbol{\Psi}\tilde{z}_r^n)||_2}{\|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}}\right).\end{aligned} \tag{25.18}$$

The ROM satisfies the energy decrease property $\mathscr{E}_h(u_{h,r}^{n+1}) \leq \mathscr{E}_h(u_{h,r}^n)$ when the right hand side of (25.18) is non-positive, i.e., if the time-step size is bounded as

$$\Delta t \leq \frac{\|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)}}{\|\boldsymbol{R}^{-1}\|_2 \|(\boldsymbol{P}^T\boldsymbol{W})^{-1}\|_2 \|(\boldsymbol{I} - \boldsymbol{W}\boldsymbol{W}^T)\boldsymbol{f}(\boldsymbol{\Psi}\tilde{\boldsymbol{z}}_r^n)\|_2}. \tag{25.19}$$

The columns of the matrix $\boldsymbol{W}$ are orthonormal and $\|(\boldsymbol{P}^T\boldsymbol{W})^{-1}\|_2$ and $\|\boldsymbol{R}^{-1}\|_2$ are of moderate size. They differ in the numerical tests between 10–30, and 30–60, respectively. The upper bound for Δt in (25.19) for each time step can be extended for all time steps to the following global upper bound:

$$\Delta t \leq \frac{\|u_{h,r}\|_{m,L^2(\Omega)}}{\|\boldsymbol{R}^{-1}\|_2 \|(\boldsymbol{P}^T\boldsymbol{W})^{-1}\|_2 \|\boldsymbol{f}(\boldsymbol{\Psi}\tilde{\boldsymbol{z}}_r)\|_{M,2}}, \tag{25.20}$$

where

$$\|u_{h,r}\|_{m,L^2(\Omega)} = \min_{1\leq n\leq J-1} \|u_{h,r}^{n+1} - u_{h,r}^n\|_{L^2(\Omega)},$$

$$\|\boldsymbol{f}(\boldsymbol{\Psi}\tilde{\boldsymbol{z}}_r)\|_{M,2} = \max_{1\leq n\leq J-1} \|(\boldsymbol{I} - \boldsymbol{W}\boldsymbol{W}^T)\boldsymbol{f}(\boldsymbol{\Psi}\tilde{\boldsymbol{z}}_r^n)\|_2.$$

The global upper bound for the time step-size Δt in the right hand side of (25.20), is a sufficiently large number so that we can choose Δt sufficiently large in the numerical examples in Sect. 25.4. Hence, the DEIM reduced energy decreases almost unconditionally for large time step-size.

25.3.4 POD Greedy Adaptive Sampling

For an efficient offline-online computation of the reduced basis functions $\{\psi_i\}_{i=1}^N$, several greedy sampling algorithms are developed for finite difference, finite element and finite volume methods (see for example [15, 19, 35]). In this section we will describe the POD-greedy sampling procedure for the SIPG discretized Allen-Cahn equation. Let $\boldsymbol{u}_{\mu^*}$ denotes the solution vector of the FOM related to a parameter value μ^*. Let us also denote by $POD_{\boldsymbol{X}}(\boldsymbol{B}, k)$ the operator on the Euclidean space with $\boldsymbol{X}$-weighted inner product ($\boldsymbol{X}$ is a positive definite matrix), and giving k POD basis functions of the matrix $\boldsymbol{B}$, related to the first k largest singular values. For instance, for a parameter value μ^*, the $\boldsymbol{M}$-orthonormal reduced modes $\{\Psi_i\}_{i=1}^N$, coefficient vectors of the L^2-orthonormal reduced basis functions $\{\psi_i\}_{i=1}^N$, may be computed through the generalized singular value decomposition [25] of the snapshot matrix $[\boldsymbol{u}_{\mu^*}^{n_1}, \ldots, \boldsymbol{u}_{\mu^*}^J] \in \mathbb{R}^{\mathcal{N}\times J}$ by $\{\Psi_1, \ldots, \Psi_N\} = POD_{\boldsymbol{M}}([\boldsymbol{u}_{\mu^*}^{n_1}, \ldots, \boldsymbol{u}_{\mu^*}^J], N)$.

Because the computation of the POD modes for time dependent parametrized PDEs is computationally demanding, we develop an adaptive POD-greedy (PODG) algorithm, where we perform a greedy search among a parameter space $\mathscr{M}$ [19].

The algorithm starts by selecting an initial parameter set $\mathscr{M}_0 = \{\mu^*\}$, where μ^* belongs to a training set $\mathscr{M}_{train} = \{\mu_1, \ldots, \mu_{n_s}\} \subset \mathscr{M}$, and an empty reduced space $V^0_{h,r} = \{0\}$. At the k-th greedy iteration, we determine the parameter $\mu^* \in \mathscr{M}_{train}$ for which an error indicator $\Delta_k(\mu^*)$ is related to the reduced system (25.16) on $V^{k-1}_{h,r}$. Then, we extend the reduced space $V^{k-1}_{h,r}$ by adding a single POD mode corresponding to the dominant singular value of $\boldsymbol{e}_{\mu^*} := [\boldsymbol{e}^1_{\mu^*}, \ldots, \boldsymbol{e}^J_{\mu^*}]$, where $\boldsymbol{e}^n_{\mu^*} = \boldsymbol{u}^n_{\mu^*} - \text{Proj}_{V^{k-1}_{h,r}} \boldsymbol{u}^n_{\mu^*}$ is the projection error on $V^{k-1}_{h,r}$. Here, the single POD mode is computed by the operator $POD_{\boldsymbol{M}}(\boldsymbol{e}_{\mu^*}, 1)$. We stop the greedy iteration either until a predefined maximum number N_{max} is reached, or the error indicator $\Delta_k(\mu^*)$ is below a prescribed tolerance TOL_G. We use as an error indicator the residual-based a-posteriori indicator

$$\Delta_k(\mu) = \left(\Delta t \sum_{n=1}^{J} \|R_h(u^n_{r,\mu})\|_{H^{-1}} \right)^{1/2},$$

where $\|\cdot\|_{H^{-1}}$ is the dual norm on H^1, and $R_h(u^n_{r,\mu})$ denotes the residual of the reduced system (25.14) on the n-th time level after time discretization.

In addition, at each greedy iteration, the computation of the error indicators $\Delta_k(\mu_i)$, $i = 1, \ldots, n_s$, requires the solution of the reduced system (25.16) for several times. For an efficient offline/online decomposition, we make use of the affine dependence of the bilinear form a_h on the parameter ϵ. Thus the stiffness matrix $\boldsymbol{A}^1$ related to the bilinear form $a_h(1; u, v)$ is computed in the offline stage only once. In the online stage, $\boldsymbol{A}_r = \epsilon \boldsymbol{\Psi}^T \boldsymbol{A}^1 \boldsymbol{\Psi}$ is computed without an additional cost. On the other hand, the non-linear term related to the logarithmic potential does not depend affinely on the parameter θ, therefore in the online stage, the non-linear vector in (25.16) is approximated using DEIM, which requires a quite small number of operations, $M \ll \mathscr{N}$ for the nonlinear vector and $n_q M$ for the Jacobian computation. The matrix $\boldsymbol{Q}$ in the DEIM is computed only once in the offline stage. Moreover, in the POD-Greedy basis computation, we also use the DEIM approximation ("Inner DEIM" in the Algorithm 1). Related to a certain parameter value μ^*, the (temporary) inner DEIM basis functions $\{W_1, \ldots, W_M\}$ are computed through the POD of the snapshot ensemble $\mathscr{F}_* := [\boldsymbol{f}^1_{\mu^*}, \ldots, \boldsymbol{f}^J_{\mu^*}]$ of the non-linear vectors; i.e. $\{W_1, \ldots, W_M\} := POD_{\boldsymbol{I}}(\mathscr{F}_*, M)$, where $\boldsymbol{I}$ is the identity matrix. In order to minimize the error induced by DEIM, we take a sufficiently large $M = M_{max,*} \leq rank(\mathscr{F}_*)$ at each greedy iteration. Finally, we construct the (final) outer DEIM basis functions by applying the POD to the snapshot matrix collecting all the snapshots for the parameter values $\mu \in \mathscr{M}_N$, which are stored in the POD-greedy algorithm.

The solutions exhibit larger gradients for the sharp interface limit when $\epsilon \to 0$. In this case the FOM solutions are computed on spatially non-uniform grids using moving mesh methods [28] or adaptive finite elements [16, 37]. Using space adaptive methods, more points are located at the sharp interface in order to resolve the steep gradients. Compared to the fixed meshes, space-adaptive meshes require

Algorithm 1 POD-greedy algorithm

```
Input: Samples $\mathscr{M}_{train} = \{\mu_i\}$, $|\mathscr{M}_{train}| = n_s$, tolerance $TOL_G$
Output: $V_{h,r}^N := \mathrm{span}\{\Psi_1, \ldots, \Psi_N\}$, $\boldsymbol{W} := \mathrm{span}\{W_1, \ldots, W_M\}$

  $\mathscr{M}_0 := \{\mu_1\}$, $\mu^* = \mu_1$, $V_{h,r}^0 := \{0\}$, $N = 1$
  while $N \leq N_{max}$ do
    compute $\boldsymbol{u}_{\mu^*}^n$, $n = 1, 2, \ldots, J$
    set $\boldsymbol{e}_{\mu^*}^n = \boldsymbol{u}_{\mu^*}^n - \mathrm{Proj}_{V_{h,r}^{N-1}} \boldsymbol{u}_{\mu^*}^n$, $n = 1, 2, \ldots, J$
    $V_{h,r}^N \longleftarrow V_{h,r}^{N-1} \cup \mathrm{POD}_{\boldsymbol{M}}(\{\boldsymbol{e}_{\mu^*}^1, \ldots, \boldsymbol{e}_{\mu^*}^J\}, 1)$
    for $i = 1$ to $n_s$ do
      $\{W_1, \ldots, W_{M_{max,i}}\} = \mathrm{POD}_{\boldsymbol{I}}(\{\boldsymbol{f}_{\mu_i}^1, \ldots, \boldsymbol{f}_{\mu_i}^J\}, M_{max,i})$    (Inner DEIM Basis)
      solve reduced system on $V_{h,r}^N$ using DEIM approximation
      calculate error indicator $\Delta_N(\mu_i)$
    end for
    $\mu^* = \underset{\mu \in \{\mu_1, \ldots, \mu_{n_s}\}}{\mathtt{argmax}} \Delta_N(\mu)$
    if $\Delta_N(\mu^*) \leq TOL_G$ then
      $N_{max} = N$
      break
    end if
    $\mathscr{M}_N := \mathscr{M}_{N-1} \cup \{\mu^*\}$
    $N \longleftarrow N + 1$
  end while
  $\mathscr{F} \longleftarrow [\mathscr{F}_1, \ldots, \mathscr{F}_N]$,  $\mathscr{F}_i = [\boldsymbol{f}_{\mu_i}^1, \ldots, \boldsymbol{f}_{\mu_i}^J]$,  $\mu_i \in \mathscr{M}_N$,  $i = 1, \ldots, N$
  $\{W_1, \ldots, W_M\} = \mathrm{POD}_{\boldsymbol{I}}(\mathscr{F}, M)$    (Outer DEIM Basis)
```

less degrees of freedom, which also would increase the speed-up of the ROMs. On the other hand, the dimension of the snapshots changes at each time step for space-adaptive meshes in contrast to the fixed size of the snapshots for fixed meshes. In order to deal with this problem, a common discretization space can be formed. But this space would be relatively high dimensional for sharp interface problems with locally varying features. In [32] formation of the fixed common discretization space is avoided without interpolating the snapshots. But, then the error of the POD reduced solutions do not satisfy the Galerkin orthogonality to the reduced space created by the adaptive snapshots. An error analysis of the POD Galerkin method for linear elliptic problems is performed in [32] and applicability of the approach is tested for 2D linear convection problems and time dependent Burger's equation. We also mention that spatially adaptive ROMs are studied for adaptive wavelets in [1] and for adaptive mixed finite elements in [36].

25.4 Numerical Results

In this section we give two numerical tests to demonstrate the effectiveness of the ROM. In the PODG algorithm, we set the tolerance $TOL_G = 10^{-3}$ and the maximum number of PODG basis functions $N_{max} = 20$. For the selected parameter values μ^* in the greedy algorithm, the FOMs are solved using linear dG elements with uniform spatial mesh size $h := \Delta x_1 = \Delta x_2$. The average number of Newton iterations was one for solving the nonlinear equations (9) at each time step. In all examples, we present the $L^2(0, T; L^2(\Omega))$ errors of the difference between the FOM and ROM solutions, and $L^\infty(0, T)$ errors of the difference between the discrete energies, i.e. the maximum error among the discrete time instances.

25.4.1 Allen-Cahn Equation with Quartic Potential Functional

We consider the 2D Allen-Cahn equation in [21] with a quartic potential functional (25.4a), so the bistable non-linear function $f(u) = u^3 - u$, under homogeneous Neumann boundary conditions in the spatial domain $\Omega = [0, 1]^2$ and in the time interval $t \in [0, 1]$. The spatial and temporal step sizes are taken as $h = 0.015$ and $\Delta t = 0.01$, respectively. The initial condition is

$$u(x, 0) = \tanh\left(\frac{0.25 - \sqrt{(x_1 - 0.5)^2 + (x_2 - 0.5)^2}}{\sqrt{2}\epsilon}\right).$$

The FOM becomes stiff for smaller ϵ. The training set for $\mu = 1/\epsilon$ is chosen by Clenshaw-Curtis points [14] using more points in direction of larger μ, or smaller ϵ, respectively:

$$\mathcal{M}_{train} = \{10.00, 24.78, 67.32, 132.5, 212.46, 297.54, 377.5, 442.68, 485.22, 500.00\}.$$

The decrease of the error indicator and energy decreases are given in Fig. 25.1. The error plot between FOM and PODG-DEIM solutions at the final time in Fig. 25.2 shows that the dynamic of the system can be captured efficiently by 20 PODG and

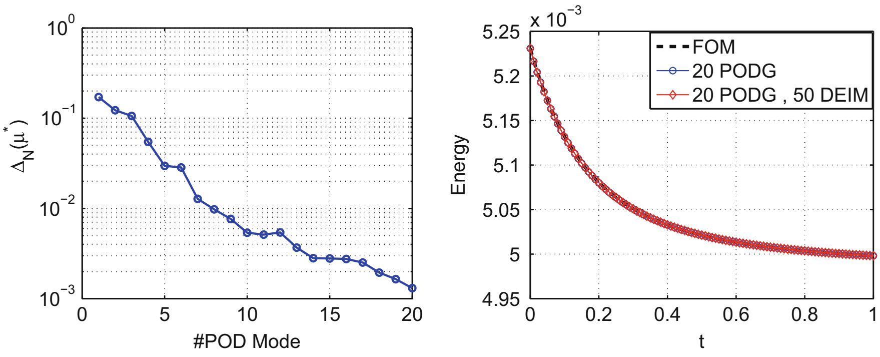


Fig. 25.1 Allen-Cahn with quartic potential: error indicator vs POD modes (*left*) and decrease of energies for $\mu = 200$ (*right*)

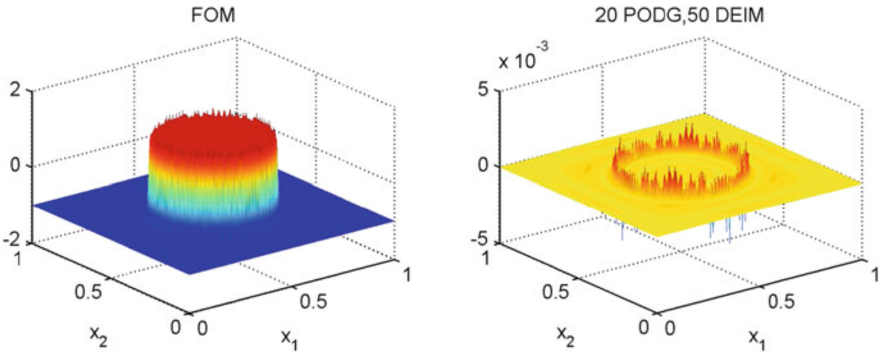


Fig. 25.2 Allen-Cahn with quartic potential: FOM profile (*left*) and error plot between FOM and PODG-DEIM (*right*) solutions, at the final time for $\mu = 200$

50 DEIM modes, which shows that FOM and ROM solutions of the Allen-Cahn equation is robust with respect to ϵ.

25.4.2 Allen-Cahn Equation with Logarithmic Potential Functional

Our second example is the 2D Allen-Cahn equation in [30], obtained here through a scale of the system (25.1) by a factor $\sqrt{\epsilon}/2$. We consider the system under periodic boundary conditions, and with a non-convex logarithmic potential (25.4b). We work on $\Omega = [0, 2\pi]^2$ with the terminal time $T = 1$. For the mesh sizes, we take $\Delta t = 0.01$ and $h \approx 0.015$. We accept the initial condition as $u(x, 0) = 0.05(2\times\texttt{rand}-1)$, where the term rand stands for a random number in [0, 1].

In the PODG algorithm, we choose now the temperature as a parameter by setting $\mu = \theta$ and we fix $\epsilon = 0.04$. The training set $\mathcal{M}_{train}$ is taken as the set consisting of the elements $\mu_k = 0.05 + 0.03(k - 1) \subset [0.05, 0.17]$, $k = 1, \ldots, 5$. The decrease of the error indicator is given in Fig. 25.3, left. The solution profile and error plot between FOM and PODG-DEIM solutions at the final time are presented in Fig. 25.4, using 20 PODG and 50 DEIM modes. In Table 25.1, we give the numerical errors for the solutions and energies between FOM and ROM solutions for both examples.

Finally, in Table 25.2, we present the CPU times and speed-up factors related to the solutions of FOM, PODG without DEIM and PODG-DEIM. It can be easily seen that PODG with DEIM dramatically improves the computational efficiency.

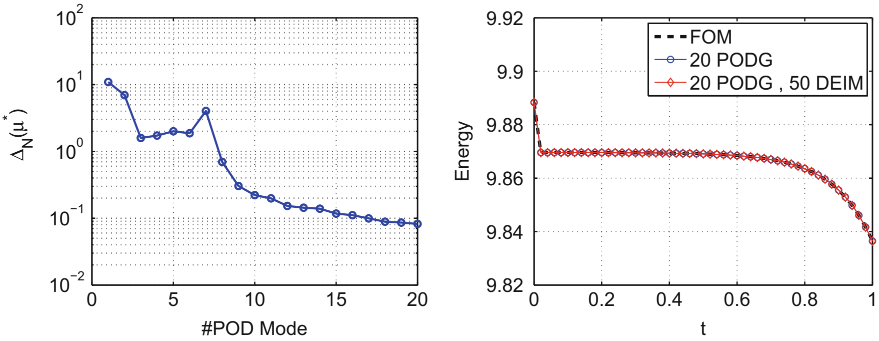


Fig. 25.3 Allen-Cahn with logarithmic potential: error indicator vs POD modes (*left*) and decrease of energies for $\theta = 0.10$ (*right*)

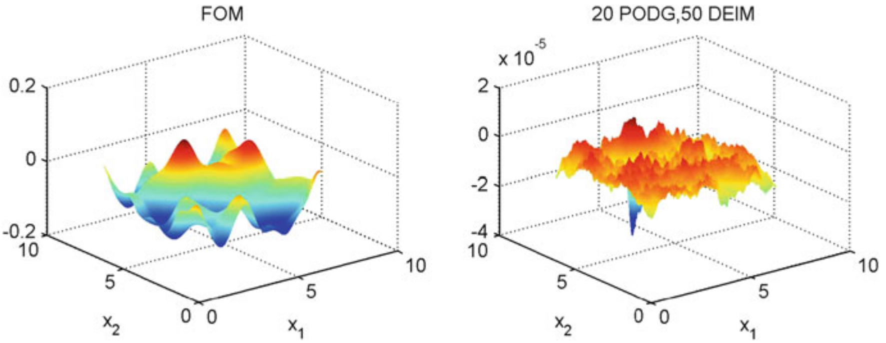


Fig. 25.4 Allen-Cahn with logarithmic potential: FOM profile (*left*) and error plot between FOM and PODG-DEIM (*right*) solutions, at the final time for $\theta = 0.10$

Table 25.1 FOM-ROM solution errors and errors between discrete energies

	Quartic F		Logarithmic F	
	Solution	Energy	Solution	Energy
PODG	9.87e-05	1.63e-06	9.66e-06	7.72e-07
PODG-DEIM	9.94e-05	1.64e-06	1.08e-05	2.43e-06

Table 25.2 CPU times and speed-up factors

	Wall clock time (s)			Speed-up factor	
	FOM	PODG	PODG-DEIM	PODG	PODG-DEIM
AC (quartic F)	50.64	9.72	2.23	5.21	22.71
AC (logarithmic F)	40.49	7.33	1.78	5.52	22.75

Acknowledgements The authors would like to thank the reviewer for the comments and suggestions that helped to improve the manuscript.

References

1. Ali, M., Urban, K.: Reduced basis exact error estimates with wavelets. In: Karasözen, B., Manguğlu, M., Tezer-Sezgin, M., Göktepe, S., Uğur, Ö. (eds.) Numerical Mathematics and Advanced Applications ENUMATH 2015, Lecture Notes in Computational Science and Engineering, vol. 112, pp. 359–367. Springer International Publishing, Switzerland (2016)
2. Allen, M.S., Cahn, J.W.: A microscopic theory for antiphase boundary motion and its application to antiphase domain coarsening. Acta Metall. **27**(6), 1085–1095 (1979)
3. Antil, H., Heinkenschloss, M., Sorensen, D.C.: Application of the discrete empirical interpolation method to reduced order modeling of nonlinear and parametric systems. In: Quarteroni, A., Rozza, G. (eds.) Reduced Order Methods for Modeling and Computational Reduction, MS and A. Model. Simul. Appl., vol. 9, pp. 101–136. Springer Italia, Milan (2014)
4. Arnold, D.N.: An interior penalty finite element method with discontinuous elements. SIAM J. Numer. Anal. **19**, 724–760 (1982)
5. Arnold, D., Brezzi, F., Cockborn, B., Marini, L.: Unified analysis of discontinuous Galerkin methods for elliptic problems. SIAM J. Numer. Anal. **39**, 1749–1779 (2002)
6. Barrault, M., Maday, Y., Nguyen, N.C., Patera, A.T.: An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. C. R. Math. **339**(9), 667–672 (2004)
7. Barrett, J.W., Blowey, J.F.: Finite element approximation of the Cahn–Hilliard equation with concentration dependent mobility. Math. Comput. **68**(226), 487–517 (1999)
8. Benner, P., Gugercin, S., Willcox, K.: A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev. **57**(4), 483–531 (2015)
9. Celledoni, E., Grimm, V., McLachlan, R., McLaren, D., O'Neale, D., Owren, B., Quispel, G.: Preserving energy resp. dissipation in numerical {PDEs} using the "average vector field" method. J. Comput. Phys. **231**(20), 6770–6789 (2012)
10. Chaturantabut, S., Sorensen, D.C.: A state space error estimate for POD–DEIM nonlinear model reduction. SIAM J. Numer. Anal. **50**(1), 46–63 (2012)
11. Chaturantabut, S., Sorensen, D.C.: Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. **32**(5), 2737–2764 (2010)
12. Choi, J.W., Lee, H.G., Jeong, D., Kim, J.: An unconditionally gradient stable numerical method for solving the Allen–Cahn equation. Phys. A Stat. Mech. Appl. **388**(9), 1791–1803 (2009)
13. Christlieb, A., Jones, J., Promislow, K., Wetton, B., Willoughby, M.: High accuracy solutions to energy gradient flows from material science models. J. Comput. Phys. **257**, 193–215 (2014)
14. Clenshaw, C.W., Curtis, A.R.: A method for numerical integration on an automatic computer. Numer. Math. **2**(1), 197–205 (1960)
15. Drohmann, M., Haasdonk, B., Ohlberger, M.: Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. SIAM J. Sci. Comput. **34**(2), A937–A969 (2012)
16. Feng, X., Wu, H.j.: A posteriori error estimates and an adaptive finite element method for the Allen–Cahn equation and the mean curvature flow. J. Sci. Comput. **24**(2), 121–146 (2005)
17. Feng, X., Song, H., Tang, T., Yang, J.: Nonlinear stability of the implicit–explicit methods for the Allen–Cahn equation. Inverse Problems Imaging **7**(3), 679–695 (2013)
18. Feng, X., Tang, T., Yang, J.: Stabilized Crank-Nicolson/Adams-Bashforth schemes for phase field models. East Asian J. Appl. Math. **3**, 59–80 (2013)
19. Grepl, M.A.: Model order reduction of parametrized nonlinear reaction-diffusion systems. Comput. Chem. Eng. **43**, 33–44 (2012)

20. Grepl, M.A., Maday, Y., Nguyen, N.C., Patera, A.T.: Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. ESAIM: M2AN **41**(3), 575–605 (2007)
21. Guo, R., Ji, L., Xu, Y.: High order local discontinuous Galerkin methods for the Allen–Cahn equation: analysis and simulation. J. Comput. Math. **34**, 135–158 (2016)
22. Hairer, E.: Energy-preserving variant of collocation methods. J. Numer. Anal. Ind. Appl. Math. **5**, 73–84 (2010)
23. Ionita, A.C., Antoulas, A.C.: Data-driven parametrized model reduction in the loewner framework. SIAM J. Sci. Comput. **36**(3), A984–A1007 (2014)
24. Kalashnikova, I., Barone, M.F.: Efficient non-linear proper orthogonal decomposition/Galerkin reduced order models with stable penalty enforcement of boundary conditions. Int. J. Numer. Methods Eng. **90**(11), 1337–1362 (2012)
25. Karasözen, B., Kücükseyhan, T., Uzunca, M.: Structure preserving integration and model order reduction of skew-gradient reaction-diffusion systems. Ann. Oper. Res. 1–28 (2015)
26. Liu, F., Shen, J.: Stabilized semi-implicit spectral deferred correction methods for Allen–Cahn and Cahn–Hilliard equations. Math. Methods Appl. Sci. **38**(18), 4564–4575 (2013)
27. Rivière, B.: Discontinuous Galerkin methods for solving elliptic and parabolic equations, Theory and implementation. SIAM (2008)
28. Shen, J., Yang, X.: An efficient moving mesh spectral method for the phase-field model of two-phase flows. J. Comput. Phys. **228**(8), 2978–2992 (2009)
29. Shen, J., Yang, X.: Numerical approximations of Allen-Cahn and Cahn-Hilliard equations. Discret. Contin. Dyn. Syst. A **28**, 1669–1691 (2010)
30. Shen, J., Tang, T., Yang, J.: On the maximum principle preserving schemes for the generalized Allen–Cahn equation. Commun. Math. Sci. **14**, 1517–1534 (2016)
31. Song, H., Jiang, L., Li, Q.: A reduced order method for Allen–Cahn equations. J. Comput. Appl. Math. **292**, 213–229 (2016)
32. Ullmann, S., Rotkvic, M., Lang, J.: POD-Galerkin reduced-order modeling with adaptive finite element snapshots. J. Comput. Phys. **235**, 244–258 (2016)
33. Vemaganti, K.: Discontinuous Galerkin methods for periodic boundary value problems. Numerical Methods Partial Differ. Equ. **23**(3), 587–596 (2007)
34. Volkwein, S.: Optimal control of a phase-field model using proper orthogonal decomposition. ZAMM - J. Appl. Math. Mech. / Zeitschrift für Angewandte Mathematik und Mechanik **81**(2), 83–97 (2001)
35. Wirtz, D., Sorensen, D.C., Haasdonk, B.: A posteriori error estimation for DEIM reduced nonlinear dynamical systems. SIAM J. Sci. Comput. **36**(2), A311–A338 (2014)
36. Yano, Masayuki: A minimum-residual mixed reduced basis method: exact residual certification and simultaneous finite-element reduced-basis refinement. ESAIM: M2AN **50**(1), 163–185 (2016)
37. Zhang, J., Du, Q.: Numerical studies of discrete approximations to the Allen–Cahn equation in the sharp interface limit. SIAM J. Sci. Comput. **31(4)**, 3042–3063 (2009)

Chapter 26
MOR-Based Uncertainty Quantification in Transcranial Magnetic Stimulation

Lorenzo Codecasa, Konstantin Weise, Luca Di Rienzo, and Jens Haueisen

Abstract Field computation for Transcranial Magnetic Stimulation requires the knowledge of the electrical conductivity profiles in the human head. Unfortunately, the conductivities of the different tissue types are not exactly known and vary from person to person. Consequently, the computation of the electric field in the human brain should incorporate the uncertainty in the conductivity values. In this paper, we compare a non-intrusive polynomial chaos expansion and a new intrusive parametric Model Order Reduction approach for the sensitivity analysis in Transcranial Magnetic Stimulation computations. Our results show that compared to the non-intrusive method, the new intrusive method provides similar results but shows two orders of magnitude reduced computation time. We find monotonically decreasing errors for increasing state-space dimensions, indicating convergence of the new method. For the sensitivity analysis, both Sobol coefficients and sensitivity coefficients indicate that the uncertainty of the white matter conductivity has the largest influence on the uncertainty in the field computation, followed by gray matter and cerebrospinal fluid. Consequently, individual white matter conductivity values should be used in Transcranial Magnetic Stimulation field computations.

L. Codecasa • L. Di Rienzo (✉)
Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, Milano, Italy
e-mail: lorenzo.codecasa@polimi.it; luca.dirienzo@polimi.it

K. Weise
Advanced Electromagnetics Group, Technische Universitaet Ilmenau, Ilmenau, Germany

Max Planck Institute for Human Cognitive and Brain Sciences, Leipzig, Germany
e-mail: konstantin.weise@tu-ilmenau.de; kweise@cbs.mpg.de

J. Haueisen
Institute of Biomedical Engineering and Informatics, Technische Universitaet Ilmenau, Ilmenau, Germany
e-mail: jens.haueisen@tu-ilmenau.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_26

26.1 Introduction

Transcranial Magnetic Stimulation (TMS) is a non-invasive technique to stimulate cortical regions of the human brain by the principle of electromagnetic induction [1, 2]. The complex geometry of the human brain requires the application of numerical techniques such as the Finite Element Method (FEM) to compute the spatial distribution of the induced electric field [12]. The knowledge of the electrical conductivity of the biological tissues constitutes one of the main elements to predict the induced electric field inside the human brain since they directly affect the solution of the boundary value problem under consideration. However, *in vivo* measurements of those tissue parameters are difficult to obtain and vary between subjects, which make exact individual assertions currently impossible [6]. Preinvestigations in [8, 18] underlined the necessity to perform an extended uncertainty and sensitivity analysis in this framework using a realistic head model and there is an essential need for more effective techniques due to the increasing model complexity of high-resolution realistic head models. In most cases, Monte Carlo (MC) methods are too expensive and techniques based on Polynomial Chaos Expansion (PCE) are favourable. In [4] a novel approach based on Parametric Model Order Reduction (PMOR) was proposed which allows to reduce the computational costs of PCE approaches. Here we add sensitivity analysis to the work presented in [4]. Recent works (e.g. [5]) propose somehow similar approaches, but for different applications. In [14] model reduction and sensitivity analysis are summarized.

26.2 Methods

26.2.1 TMS Deterministic Modeling

In the deterministic modeling phase we use a realistic head model [19], which is shown in Fig. 26.1. It contains five different tissues, namely scalp, skull, cerebrospinal fluid (CSF), grey matter (GM) and white matter (WM). The model is discretized using approximately $N = 2.8 \cdot 10^6$ linear tetrahedral finite elements. The coil[1] is approximated by 2712 magnetic dipoles constituted in three layers [16]. The electromagnetic problem at hand is simplified due to the low electrical conductivities and moderate excitation frequencies which are in the range of 2–3 kHz so that the secondary magnetic field from the induced eddy currents can be neglected [12]. In this way, the magnetic field can be expressed in terms of the magnetic vector potential $\mathbf{a}_c$ produced by the excitation coil ($\mathbf{b}_c = \nabla \times \mathbf{a}_c$, $\nabla \cdot \mathbf{a}_c = 0$). Considering the current conservation law, this reduces to solve

[1] The excitation coil is a Magstim 70 mm double coil with 9 windings which is placed above the motor cortex area M1 (Brodman area 4) at a distance of 4 mm from the skalp.

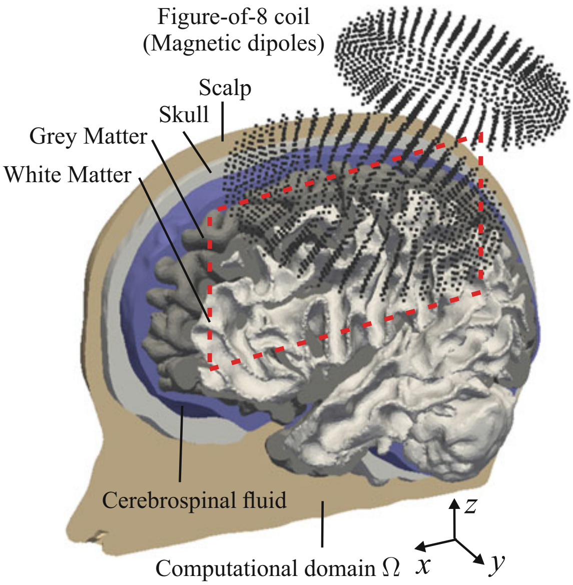


Fig. 26.1 FEM model of the human head used for uncertainty quantification

the following equation at angular frequency ω with Neumann conditions on the boundary $\partial\Omega$ of the spatial domain Ω

$$\nabla \cdot (-\sigma(\mathbf{r},\mathbf{p})\nabla\varphi(\mathbf{r},\mathbf{p})) = i\omega\,\nabla \cdot (\sigma(\mathbf{r},\mathbf{p})\mathbf{a}_c(\mathbf{r})), \tag{26.1}$$

in which the unknown $\varphi(\mathbf{r},\mathbf{p})$ is the electric potential, $\mathbf{a}_c(\mathbf{r})$ is the known magnetic vector potential, and $\sigma(\mathbf{r},\mathbf{p})$ is the electric conductivity; the latter can be assumed to be a *linear* combination of the P parameters p_i, forming vector $\mathbf{p}$

$$\sigma(\mathbf{r},\mathbf{p}) = \sigma_0(\mathbf{r}) + \sum_{i=1}^{P} \sigma_i(\mathbf{r})\,p_i. \tag{26.2}$$

As it is well known, the FEM discretization is achieved by rewriting the electromagnetic problem in the weak form (problem $\mathscr{D}$)

$$\int_\Omega \nabla\varphi'(\mathbf{r}) \cdot \sigma(\mathbf{r},\mathbf{p})\nabla\varphi(\mathbf{r},\mathbf{p})\,d\mathbf{r} = i\omega \int_\Omega \nabla\varphi'(\mathbf{r}) \cdot \sigma(\mathbf{r},\mathbf{p})\mathbf{a}_c(\mathbf{r})\,d\mathbf{r} \tag{26.3}$$

for all functions $\varphi'(\mathbf{r})$, in which both $\varphi'(\mathbf{r})$ and $\varphi(\mathbf{r},\mathbf{p})$ belong to the linear tetrahedral finite element space $\mathscr{X}$ of dimension $|\mathscr{X}| = N$ whose Degrees of Freedom form vector $\boldsymbol{x}(\mathbf{p})$.

26.2.2 TMS Stochastic Modeling

The electrical conductivities of scalp and skin are modelled as deterministic since they hardly affect the induced electric field inside the human brain. On the other hand, the conductivities of CSF, GM, and WM show a wide spread across individuals and measurements and are then modelled as uniform distributed random variables. The conductivity values are defined as in Table 26.1.

In a PCE analysis, the electric potential $\varphi(\mathbf{r}, \mathbf{p})$ is approximated in the form

$$\varphi(\mathbf{r}, \mathbf{p}) = \sum_{|\boldsymbol{\alpha}|\leq Q} \varphi_{\boldsymbol{\alpha}}(\mathbf{r})\psi_{\boldsymbol{\alpha}}(\mathbf{p}), \tag{26.4}$$

in which $\boldsymbol{\alpha}$ are multi-indices of P elements and $\psi_{\boldsymbol{\alpha}}(\mathbf{p})$ are polynomials of degrees less than a chosen Q, forming an orthonormal basis in the probability space of random variable p_i, given by

$$\psi_{\boldsymbol{\alpha}}(\mathbf{p}) = \prod_{i=1}^{P} l_{\alpha_i}(p_i), \tag{26.5}$$

with $l_{\alpha_i}(p_i)$ being orthonormal Legendre polynomials.

The number of coefficients in a maximum order PCE is given by

$$M = \binom{P+Q}{P}. \tag{26.6}$$

Both intrusive and non-intrusive approaches to PCE can be used. Non-intrusive approaches are commonly adopted as the most efficient alternatives to Monte Carlo technique. In this way the coefficients $\varphi_{\boldsymbol{\alpha}}(\mathbf{r})$ are determined from the solutions $\varphi(\mathbf{r}, \mathbf{p})$ of the deterministic problems $\mathscr{D}$ for all values of $\mathbf{p}$ in a proper set $\mathscr{G}$. However, even using sparse-grids [20], the set $\mathscr{G}$ becomes very large when the number of parameters P or the polynomial degree Q increases. Thus the number of deterministic problems to be solved becomes infeasible.

Table 26.1 Deterministic and stochastic conductivities $[S/m]$

Tissue	Deterministic or stochastic	Lower value	Upper value	References
Scalp	Deterministic	0.34	0.34	[9]
Skull	Deterministic	0.025	0.025	[10, 15]
CSF	Stochastic	1.432	2.148	[3]
Grey matter	Stochastic	0.153	0.573	[6, 7, 11]
White matter	Stochastic	0.094	0.334	[6, 7, 11]

26.2.3 Parametric Model Order Reduction Approach

Hereafter we propose Algorithm 1 alternative to the non-intrusive PCE approach [18], based on the well-known greedy algorithm [19], which constructs a reduced order model tailored to PCE analysis solving a much smaller number of deterministic problems with respect to the non-intrusive PCE approaches. Moreover the computational cost of the solution to these deterministic problems is much smaller with respect to the non-intrusive approaches, since accurate estimations of the solution to these deterministic problems are derived from the reduced order model solutions taken as starting points in the adopted iterative methods for solving linear systems. The PCE expansion of the solution to the original problem is then obtained from such reduced order model.

In the algorithm, at step 1, the FEM discretization of the electromagnetic deterministic problem (26.3) is solved for each selected value of $\mathbf{p}$. A preconditioned conjugate gradient method is used and the number of iterations is reduced by assuming as initial point the $\hat{\varphi}(\mathbf{r}, \mathbf{p})$ estimation provided by the previously computed compact model. At step 2 an orthonormal basis of space $\mathscr{S}_k$ is generated, computing a set of functions $v_j(\mathbf{r})$, with $j = 1, \ldots, k$, generating all functions $\varphi(\mathbf{r}, \mathbf{p})$ computed at step 1 and forming the column vector

$$\boldsymbol{v}(\mathbf{r}) = [v_j(\mathbf{r})]. \tag{26.7}$$

Algorithm 1 PMOR-based algorithm

```
Set $k := 0$ (dimension of the reduced model)
Set $\vartheta := +\infty$ (norm of the residual)
Set linear space $\mathscr{S}_0 := \emptyset$
Choose vector $\mathbf{p}$ in $\mathscr{G}$
Set $\hat{\varphi}(\mathbf{r}, \mathbf{p}) := 0$
while $\vartheta > \varepsilon$ do
    Set $k := k + 1$
1      Solve problem (26.3) for $\varphi(\mathbf{r}, \mathbf{p})$ using $\hat{\varphi}(\mathbf{r}, \mathbf{p})$ as initial estimation
2      Generate an orthonormal basis of the linear space $\mathscr{S}_k$ spanned by $\mathscr{S}_{k-1}$ and $\varphi(\mathbf{r}, \mathbf{p})$
3      Generate reduced order model $\mathscr{R}_k(\mathbf{p})$, projecting problem $\mathscr{D}$ onto space $\mathscr{S}_k$
    for all $\mathbf{q} \in \mathscr{G}$ do
4          Solve the reduced order model $\mathscr{R}_k(\mathbf{q})$ obtaining $\hat{\varphi}(\mathbf{r}, \mathbf{q})$ as an approximation for $\varphi(\mathbf{r}, \mathbf{q})$
5          Estimate the approximation residual $\eta$
        if $\eta > \vartheta$ then
            Set $\vartheta := \eta$
6              Set $\mathbf{p} := \mathbf{q}$
Set $K := k$
7 Determine the PCE expansion of the solution to the reduced order model $\mathscr{R}_K(\mathbf{p})$ and reconstruct the PCE expansion of $\varphi(\mathbf{r}, \mathbf{p})$
```

At step 3 the reduced order model $\mathscr{R}_k(\mathbf{p})$ is constructed. This model is obtained from (26.3) assuming that the $\mathscr{X}$ space is substituted by its subspace, spanned by functions $v_j(\mathbf{r})$, with $j = 1, \ldots, k$. In this way the compact model takes the form

$$\left(\hat{\mathbf{S}}_0 + \sum_{i=1}^{P} p_i \hat{\mathbf{S}}_i\right) \hat{\boldsymbol{x}}(\mathbf{p}) = i\omega \left(\hat{\mathbf{u}}_0 + \sum_{i=1}^{P} p_i \hat{\mathbf{u}}_i\right) \tag{26.8}$$

in which $\hat{\mathbf{S}}_i$, with $i = 1, \ldots, P$, are square matrices of dimension k given by

$$\hat{\mathbf{S}}_i = \left[\int_\Omega \nabla v_j(\mathbf{r}) \cdot \sigma_i(\mathbf{r}) \nabla v_l(\mathbf{r})\, d\mathbf{r}\right], \quad i = 0, \ldots, P, \tag{26.9}$$

and $\hat{\mathbf{u}}_i$, with $i = 0, \ldots, P$, are column vectors of k rows

$$\hat{\mathbf{u}}_i = \left[\int_\Omega \nabla v_j(\mathbf{r}) \cdot \sigma_i(\mathbf{r}) \mathbf{a}_c(\mathbf{r})\, d\mathbf{r}\right]. \tag{26.10}$$

Vector $\hat{\boldsymbol{x}}(\mathbf{p})$ allows to approximate the solution $\varphi(\mathbf{r}, \mathbf{p})$ to (26.3) as (step 4)

$$\hat{\varphi}(\mathbf{r}, \mathbf{p}) = \sum_{j=1}^{k} \hat{x}_j(\mathbf{p}) v_j(\mathbf{r}) = \boldsymbol{v}^T(\mathbf{r}) \hat{\boldsymbol{x}}. \tag{26.11}$$

At step 5, η represents the residual when $\varphi(\mathbf{r}, \mathbf{q})$ is substituted by $\hat{\varphi}(\mathbf{r}, \mathbf{q})$ in (26.3). It is worth recalling here that there are error estimators, which allow an offline-online decomposition and thus an estimation of the residual at a computational cost that scales with the dimension of the reduced space (except for the necessary computation of the respective Riesz representations of course [17]). At step 6, the value of $\mathbf{q}$ in $\mathscr{G}$ maximizing the value of η becomes the candidate $\mathbf{p}$ for solving the deterministic problem (26.3) at next step 1. At step 7, an intrusive PCE approach is applied to the reduced order model $\mathscr{R}_K$. Thus $\hat{\boldsymbol{x}}(\mathbf{p})$ is approximated by its PCE

$$\hat{\boldsymbol{x}}(\mathbf{p}) = \sum_{|\boldsymbol{\alpha}| \leq Q} \hat{\boldsymbol{y}}_{\boldsymbol{\alpha}} \psi_{\boldsymbol{\alpha}}(\mathbf{p}). \tag{26.12}$$

Substituting this expansion into (26.8), multiplying then (26.8) by $\psi_{\boldsymbol{\beta}}(\mathbf{p})$, with $|\boldsymbol{\beta}| \leq M$, and applying the expected value operator $\mathbf{E}[\cdot]$, it results in

$$\left(\mathbf{1}_M \otimes \hat{\mathbf{S}}_0 + \sum_{i=1}^{P} \mathbf{P}_i \otimes \hat{\mathbf{S}}_i\right) \mathrm{vec}(\hat{\boldsymbol{Y}}) = \left(\mathbf{e}_1 \otimes \hat{\mathbf{u}}_0 + \sum_{i=1}^{P} \mathbf{P}_i \mathbf{e}_1 \otimes \hat{\mathbf{u}}_i\right), \tag{26.13}$$

in which

$$\mathbf{P}_i = \left[\mathbf{E}[p_i \psi_{\boldsymbol{\alpha}}(\mathbf{p}) \psi_{\boldsymbol{\beta}}(\mathbf{p})]\right] \tag{26.14}$$

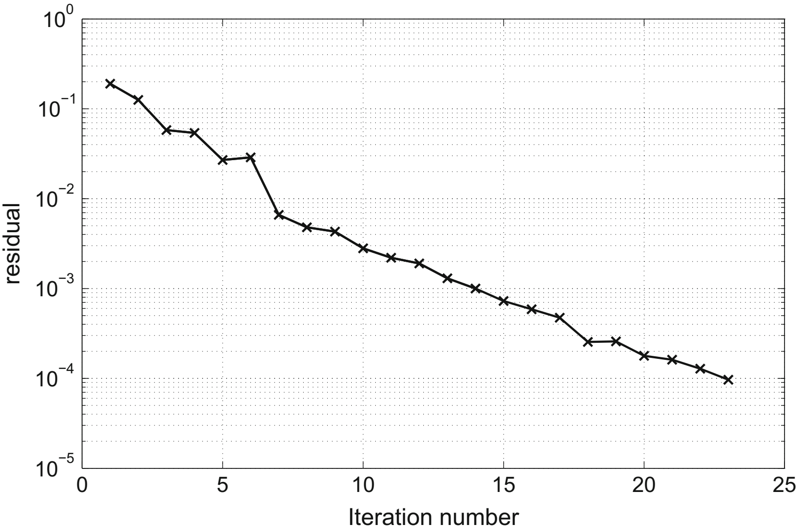


Fig. 26.2 Convergence of the algorithm represented as the residual vs. the reduced basis dimension

are square matrices of order M, $\otimes$ indicates Kronecker's tensor product, $\hat{\boldsymbol{Y}} = [\hat{\boldsymbol{y}}_\alpha]$ is a $K \times M$ matrix, $\text{vec}(\hat{\boldsymbol{Y}})$ is the vector of the stacked columns of $\hat{\boldsymbol{Y}}$, and $\mathbf{e}_1$ is a column vector of M rows made of all zeros except the first element that is one. In these and all the following definitions of matrices and vectors the entries are indexed by organizing multi-indices in lexicographic order.

This linear system of equations in the unknowns $\text{vec}(\hat{\boldsymbol{Y}})$ has reduced dimension with respect to that of the standard intrusive PCE approach, so that it can be solved at negligible cost. From the PCE expansion of $\hat{\boldsymbol{x}}(\mathbf{p})$, the PCE expansion of $\hat{\varphi}(\mathbf{r}, \mathbf{p})$ approximating the PCE of $\varphi(\mathbf{r}, \mathbf{p})$ is straightforwardly obtained as

$$\hat{\varphi}(\mathbf{r}, \mathbf{p}) = \sum_{|\boldsymbol{\beta}| \leq Q} \boldsymbol{v}(\mathbf{r}) \hat{\boldsymbol{y}}_\alpha \psi_\alpha(\mathbf{p}). \tag{26.15}$$

As can be noted in Fig. 26.2 the convergence of the algorithm is of exponential type.

26.2.4 The Non-intrusive Approach

In order to compare the numerical results of the new method the PCE-coefficients $\varphi_\alpha(\mathbf{r})$ are also determined using a traditional non-intrusive approach based on Regression (REG). The implementation presented in Weise et al. [18] is used.

In such approach the computational grid $\mathscr{G}$ is constructed as the tensor product of the roots of the Q-th order Legendre polynomials resulting in a total number of $G = Q^P$. In this way the values $\mathbf{p}_{\boldsymbol{\beta}}$ of the parameter vector are considered, in which multi-index $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_P)$, with $\beta_i = 1, \ldots, Q$ and $i = 1, \ldots, P$.

The solutions $\varphi(\mathbf{r}, \mathbf{p}_{\boldsymbol{\beta}})$ of the deterministic problems (26.3) are then computed for all the chosen values $\mathbf{p}_{\boldsymbol{\beta}}$ of the parameter vector. The N Degrees of Freedom (DoF) of each of these solutions, forming vector $\boldsymbol{x}(\mathbf{p}_{\boldsymbol{\beta}})$, define the $N \times G$ matrix $\boldsymbol{X} = [\boldsymbol{x}(\mathbf{p}_{\boldsymbol{\beta}})]$. The PCE of the DoF, forming the $N \times M$ matrix $\boldsymbol{Y} = [\boldsymbol{y}_{\boldsymbol{\alpha}}]$ are obtained solving the *overdetermined* system of equations in the least squares sense

$$\boldsymbol{Y}\mathbf{A} = \boldsymbol{X}, \tag{26.16}$$

in which $\mathbf{A} = [\psi_{\boldsymbol{\alpha}}(\mathbf{p}_{\boldsymbol{\beta}})]$ is an $M \times G$ matrix. From the PCE of the DoF, the PCE of the electric potential $\varphi_{\boldsymbol{\alpha}}(\mathbf{r})$ is derived.

26.2.5 Post-processing

The PCE of the magnitude $E(\mathbf{r}, \mathbf{p})$ of the induced electric field is determined in a post-processing stage in the form

$$E(\mathbf{r}, \mathbf{p}) = \sum_{|\boldsymbol{\alpha}| \leq Q} E_{\boldsymbol{\alpha}}(\mathbf{r}) \psi_{\boldsymbol{\alpha}}(\mathbf{p}). \tag{26.17}$$

Since the PCE polynomials are assumed orthonormal, from (26.17) the statistical mean $\mu_E(\mathbf{r})$ of $E(\mathbf{r}, \mathbf{p})$ is directly given by the first PCE coefficient. The standard deviation $\sigma_E(\mathbf{r})$ is calculated as the sum of the remaining squared coefficients:

$$\mu_E(\mathbf{r}) = E_{\mathbf{0}}(\mathbf{r}), \quad \sigma_E(\mathbf{r}) = \sqrt{\sum_{0 < |\boldsymbol{\alpha}| \leq Q} E_{\boldsymbol{\alpha}}^2(\mathbf{r})}. \tag{26.18}$$

In order to quantify how strongly the induced electric field is affected by the variations of the stochastic electrical conductivity of each tissue, we can perform a sensitivity analysis in two different approaches.

According to a first approach, first-order Sobol' sensitivity coefficients (not normalized to the total variance) can be introduced as in [13] and computed using (26.5) as

$$S_i(\mathbf{r}) = \mathbf{Var}\left[\mathbf{E}\left[E(\mathbf{r}, \mathbf{p}) | p_i\right]\right] = \sum_{\boldsymbol{\alpha} \in \mathscr{S}_i} E_{\boldsymbol{\alpha}}^2(\mathbf{r}), \tag{26.19}$$

where $\mathscr{S}_i = \{\boldsymbol{\alpha} | \alpha_j = 0 \text{ for all } j \neq i \text{ and } \alpha_i > 0\}$, with $i = 1, \ldots, P$.

Furthermore, according to a second approach, derivative-based sensitivity coefficients can be introduced as in [20] and estimated from (26.5) as

$$S_i'(\mathbf{r}) = \mathbf{E}\left[\frac{\partial E}{\partial p_i}(\mathbf{r},\mathbf{p})\right] = \sum_{|\boldsymbol{\alpha}|\le Q} E_{\boldsymbol{\alpha}}(\mathbf{r})\mathbf{E}\left[\frac{\partial \psi_{\boldsymbol{\alpha}}(\mathbf{p})}{\partial p_i}\right] = \sum_{\boldsymbol{\alpha}\in\mathscr{S}_i'} \sqrt{2\alpha_i+1}\,E_{\boldsymbol{\alpha}}(\mathbf{r}), \tag{26.20}$$

being $\mathscr{S}_i' = \{\boldsymbol{\alpha}|\alpha_j = 0 \text{ for all } j \neq i \text{ and } \alpha_i \text{ odd}\}$, with $i = 1, \ldots, P$.

26.3 Numerical Results

The grid $\mathscr{G}$ adopted in both PMOR and REG is composed of $G = 5^3 = 125$ nodes and the chosen polynomial degree for PCE is $P = 5$. The spatial distributions of μ_E and σ_E determined by PMOR (with $K = 14$) and REG approaches are shown in Fig. 26.3. The absolute difference between both approaches show minor deviations. The equivalence is underlined by the relative error in the energy norm which is $4.5 \cdot 10^{-6}$ for the mean and $1.7 \cdot 10^{-4}$ for the standard deviation. The mean induced electric field allows a more general interpretation of the estimated field distributions. Moreover, the standard deviation reveals areas in GM and WM where the electric field shows a wide spread as a result of the uncertain conductivity.

Since the PCE is performed in the whole brain, it is possible to determine the Probability Density Function (PDF) of E in every point by sampling the polynomials (26.17). The PDFs of three exemplary points located right under the excitation coil are evaluated and shown in Fig. 26.4 to further illustrate the agreement between the PMOR and the REG approach. For all three PDFs, the relative error between the two approaches is less than 0.2%. The small differences may originate from the sampling procedure since the PMOR and the REG approach did not share the same sample set. The PDFs illustrate how the shape and spread of the induced electric field vary in space. It can be observed that the spread is large inside the WM domain which is surrounded by two domains, namely GM and CSF, both obeying uncertain conductivities. A major strength of PMOR approach is its computational efficiency. It required 0.8 GB of memory and a total simulation time of 80 s using a MacBook Pro Early 2011 (Intel Core i7-2720QM @ 2.3 GHz with 4-cores, 8-threads and with 16 GB RAM). In contrast, REG approach required 2.2 GB and finished after 250 min on a more powerful computer (Intel Core i7-3770K @ 4.2 GHz with 4-cores, 8-threads and with 32 GB RAM). In this way, PMOR is more than 180 times faster than traditional non-intrusive methods. In conclusion, the MOR approach was run on a laptop while the non-intrusive approach was run on a much faster desktop computer. So the comparison of the two approaches is pessimistic in estimating the advantage of the novel approach.

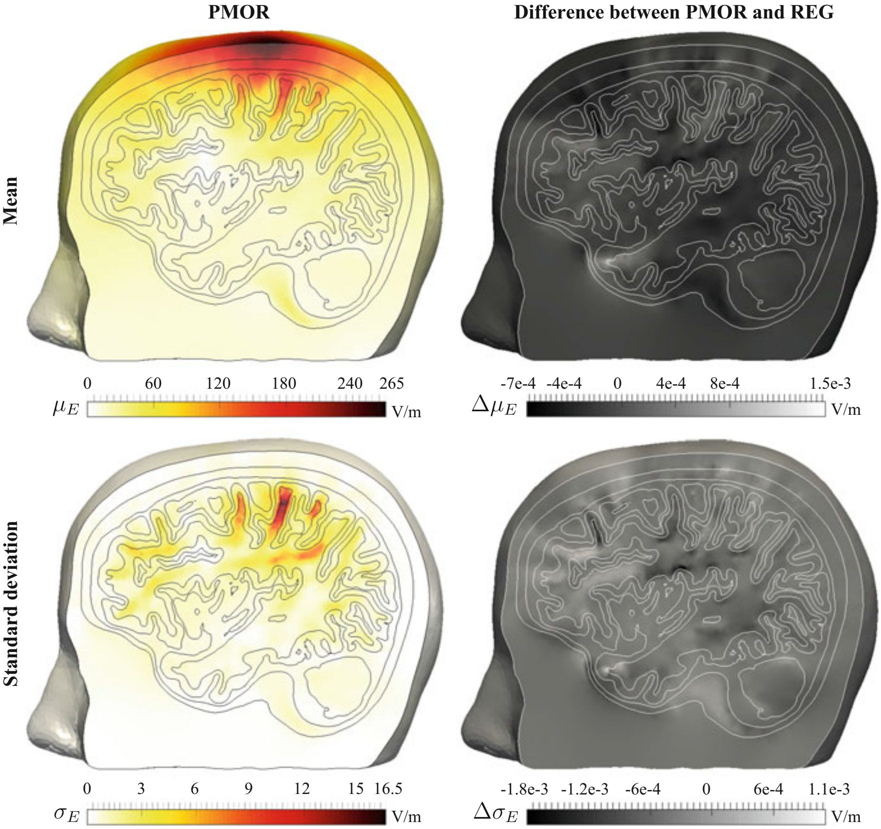


Fig. 26.3 Mean μ_E, standard deviation σ_E and absolute differences of both in the sagittal plane under the excitation coil determined by PMOR and REG

The spatial distributions of the three linear Sobol coefficients (given by (26.19)) of the biological tissues CSF, GM, and WM are computed as in [13] and are shown on the left hand side in Fig. 26.5. On the right hand side, the absolute difference between the new PMOR and the REG approach is shown. The linear Sobol coefficients quantify the contribution of the individual material conductivities to the total variance observed in the whole head with a unit of measure (V/m)^2. Consequently, they are linked to the standard deviation shown in Fig. 26.3. The absolute difference between the Sobol coefficients obtained by PMOR and REG are up to five magnitudes lower than their magnitudes indicating an excellent agreement between both approaches. In case of the Sobol coefficient for CSF, the difference plot shows an unstructured spread in the skull and scalp region, which is eventually a result of numerical noise introduced during the solving process of the linear system. It is observed from the numerical results that the linear Sobol coefficients add up to more than 99% of the total variance, indicating their prime

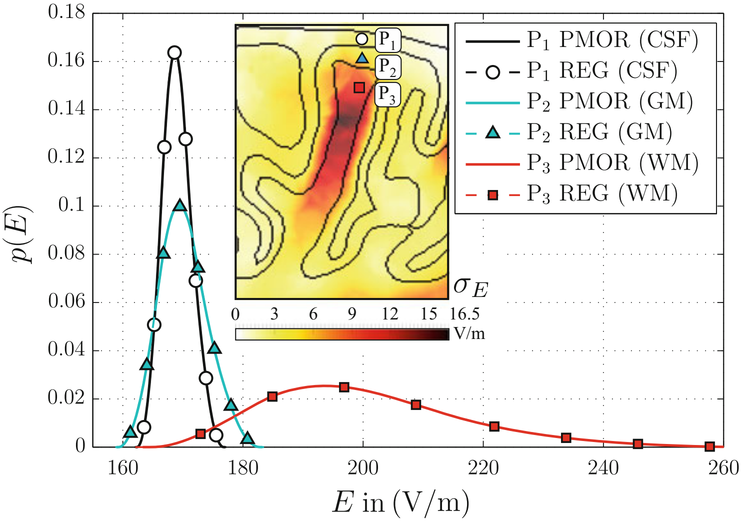


Fig. 26.4 Probability density functions of E in three points located right under the excitation coil determined by the MOR and REG. The polynomials are sampled $1 \cdot 10^6$ times in a postprocessing step

importance during this analysis. It can be seen that the spatial distributions of all three Sobol coefficients are similar while their magnitudes vary. All of them show a maximum in the white matter area at the posterior wall of the *sulcus centralis posterior* together with a sudden change to adjacent tissue areas (Fig. 26.3). In the main extend, the Sobol coefficients indicate that the whole WM region is primarily affected by the conductivity variations. It can thus be concluded that conductivity variations in GM and CSF directly affect the white matter area. Considering the magnitude of the Sobol coefficients, it can be stated that white matter contributes with up to 180 $(\text{V/m})^2$ most to the total variance of the induced electric field, directly followed by grey matter and CSF with 133 $(\text{V/m})^2$ and 60 $(\text{V/m})^2$, respectively. It is noted that this observation is in contrast to the one made during the analysis of a simplified gyrus sulcus structure in [18], since it has to be considered, that the conductivity variation of white matter differs between the present paper and [18] (here: $0.094 \leq \sigma_{\text{WM}} \leq 0.334$ S/m, [18]: $0.096 \leq \sigma_{\text{WM}} \leq 0.166$ S/m).

Next, the derivative based global sensitivity coefficients (GDS) are computed according to (26.20), which, compared to the Sobol coefficients, obey a more deterministic interpretation of sensitivity. The three GDS coefficients are shown on the left hand side in Fig. 26.6. As in the previous case, the corresponding absolute

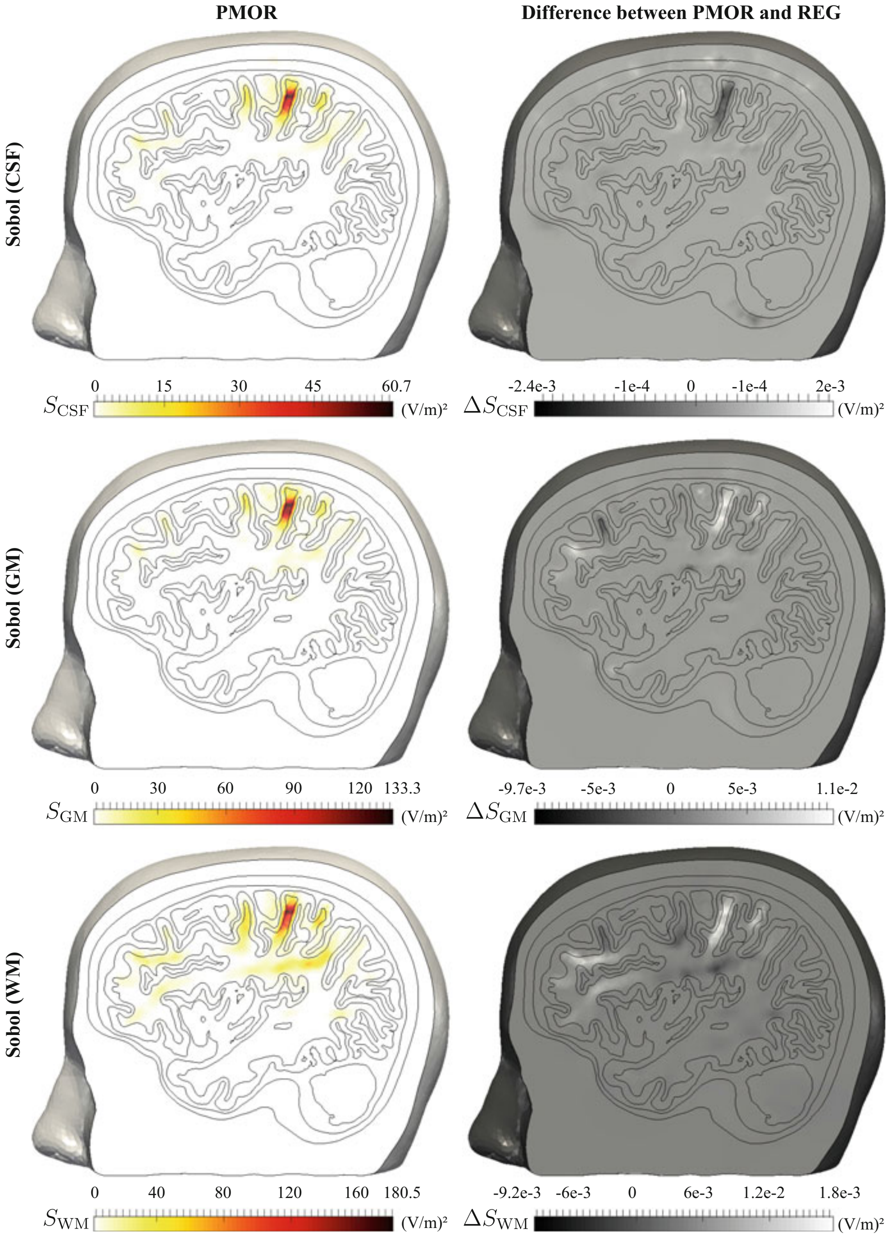

PMOR
Difference between PMOR and REG
Sobol (CSF)
0 15 30 45 60.7
S_{CSF} (V/m)²
-2.4e-3 -1e-4 0 -1e-4 2e-3
ΔS_{CSF} (V/m)²
Sobol (GM)
0 30 60 90 120 133.3
S_{GM} (V/m)²
-9.7e-3 -5e-3 0 5e-3 1.1e-2
ΔS_{GM} (V/m)²
Sobol (WM)
0 40 80 120 160 180.5
S_{WM} (V/m)²
-9.2e-3 -6e-3 0 6e-3 1.2e-2 1.8e-3
ΔS_{WM} (V/m)²

Fig. 26.5 Spatial distribution of the linear Sobol coefficients, computed as in [13]

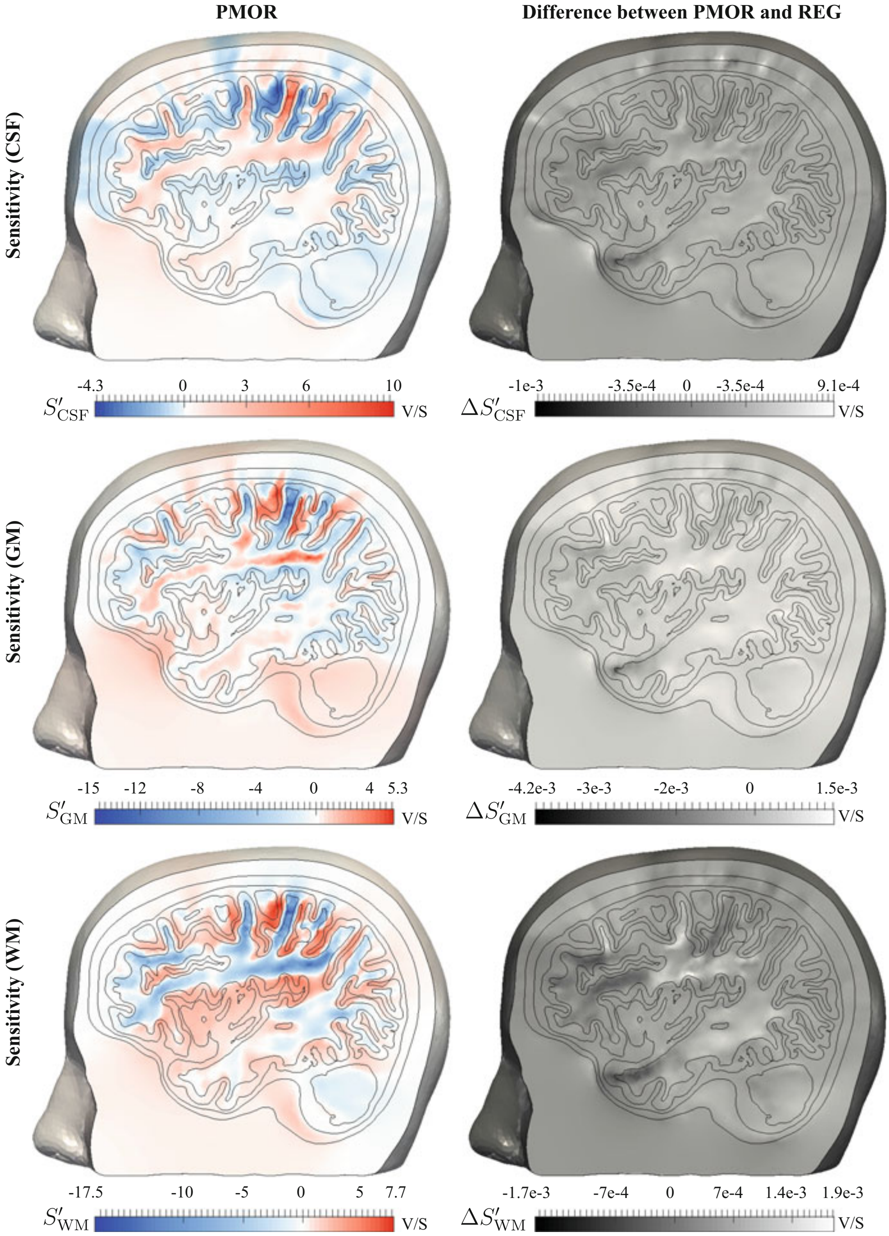

PMOR
Difference between PMOR and REG
Sensitivity (CSF)
-4.3 0 3 6 10
S'_{CSF} V/S
-1e-3 -3.5e-4 0 -3.5e-4 9.1e-4
$\Delta S'_{\mathrm{CSF}}$ V/S
Sensitivity (GM)
-15 -12 -8 -4 0 4 5.3
S'_{GM} V/S
-4.2e-3 -3e-3 -2e-3 0 1.5e-3
$\Delta S'_{\mathrm{GM}}$ V/S
Sensitivity (WM)
-17.5 -10 -5 0 5 7.7
S'_{WM} V/S
-1.7e-3 -7e-4 0 7e-4 1.4e-3 1.9e-3
$\Delta S'_{\mathrm{WM}}$ V/S

Fig. 26.6 Derivative based global sensitivity coefficients (GDS), computed as in [18]

differences between the coefficients obtained by PMOR and REG approach are shown on the right hand side. Again, both approaches show an excellent agreement and yield almost identical results. The distributions of the GDS coefficients differ strongly to the ones of the Sobol coefficients from Fig. 26.5 since they quantify the rate of change of the induced electric field with respect to a change in conductivity with a unit of measure V/S. Its sign provides information about the direction of change, whether the electric field is decreasing (minus sign, blue color) or increasing (plus sign, red color) when the conductivity rises. From an electromagnetic point of view, considering Faraday's law of induction, it is expected that, in the presence of boundaries, an increase in conductivity will lead to a drop of the electric field in the respective domain while keeping the induced current density at a constant level to fulfil the current conservation law. This can be observed by comparing the blue regions in the respective domains when comparing the individual GDS coefficients with each other. The sensitivity maps of the GDS coefficients are much broader distributed and span throughout the entire brain area compared to the Sobol coefficients. This nicely demonstrates the impact of individual conductivity variations to all other domains of the brain. However, similar to the Sobol coefficients, their magnitudes are also highest in the target region of the *sulcus centralis posterior*. As expected from electromagnetic theory, the distribution of the GDS coefficient of CSF indicates a negative, i.e. decreasing induced electric field in CSF and adjacent domains such as the considered deterministic skull and scalp region. At the same time, it can be observed that the induced electric field increases predominantly in the white matter region. This can be seen by the extended red area inside the brain and in the sulcus centralis posterior. The present observation concerning the location and the magnitude of this effect is in contrast to the one made in the simplified gyrus/sulcus model from [18]. The sensitivity maps of the GDS coefficients for GM and WM show similar patterns and magnitudes. Interestingly, the GM region at the posterior wall of the sulcus centralis is less affected by the conductivity variations, which is indicated by the white color. One reason for this could be the embedded structure of GM. However, this effect is immediately annulled in the adjacent CSF and WM region, where the induced electric field would increase or decrease, respectively, in consequence of a positive conductivity variation. Those observations are also novel compared to the ones made during the analysis of the simplified gyrus/sulcus structure [18]. Besides all differences between the spatial distributions and the unit of measure between the Sobol and the GDS coefficients, the fundamental discoveries attained by both sensitivity indicators coincide well and contribute substantially to understand the complex processes taking place in the TMS framework.

In Fig. 26.7 the convergence properties of Algorithm 1 are numerically demonstrated.

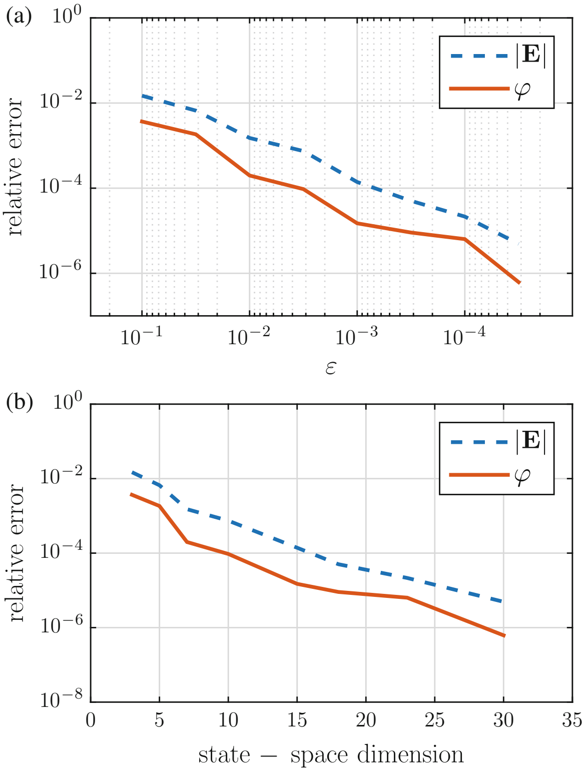


Fig. 26.7 Accuracy of the PMOR-based algorithm. (**a**) Relative error in ℓ^2-norm in both electric potential and electric field vs. the accuracy threshold ϵ of Algorithm 1. (**b**) Relative error in ℓ^2-norm in both electric potential and electric field vs. the dimension of the reduced order model

26.4 Conclusion

The present paper demonstrates the advantages and applicability of a parametric Model Order Reduction approach to uncertainty quantification in the framework of Transcranial Magnetic Stimulation modeling. Three tissue conductivities of the brain are described as random variables and the relevant statistics (mean, standard deviation, sensitivities coefficients) in the stochastic modeling of the magnetic stimulation phenomena are estimated with a dramatic reduction in the computational burden. The analysis helps to define which tissue conductivity must be most precisely be known and at which level of accuracy for an effective modeling. Besides of the first two statistical moments, the spatial distribution of different sensitivity measures are presented deepening the understanding of the complex phenomena taking place when considering uncertain conductivity data. By virtue of extending the sensitivity analysis to a realistic head model, it is possible to identify particular differences in comparison to the previous study [18], which is related to a simplified gyrus/sulcus model. The developed code is not yet optimized so we prefer not to make it available for the moment. In the light of the previous considerations, the motivation is further strengthened to extend this kind of studies also in other biomedical frameworks such as Transcranial Direct Current Stimulation in the future.

Acknowledgements The present work is partially supported by the Deutsche Forschungsgemeinschaft (DFG) under the grant WE 5985 at the Technische Universitaet Ilmenau, Germany.

References

1. Barker, A.T.: An introduction to the basic principles of magnetic nerve stimulation. J. Clin. Neurophysiol. **8**, 26–37 (1991)
2. Barker, A.T., Jalinous, R., Freeston, I.L.: Non-invasive magnetic stimulation of human motor cortex. Lancet **325**, 1106–1107 (1985)
3. Baumann, S.B., Wozny, D.R., Kelly, S.K., Meno, F.M.: The electrical conductivity of human cerebrospinal fluid at body temperature. IEEE Trans. Biomed. Eng. **44**, 220–223 (1997)
4. Codecasa, L., Di Rienzo, L., Weise, K., Gross, S., Haueisen, J.: Fast MOR-based approach to transcranial magnetic stimulation. IEEE Trans. Magn. **52**, Article: 7200904 (2016)
5. Drissaoui, M.A., Lanteri, S., Lévêque, P., Musy, F., Nicolas, L., Perrussel, R., Voyer, D.: A stochastic collocation method combined with reduced basis method to compute uncertainties in numerical dosimetry. IEEE Trans. Magn. **48**, 563–566 (2012)
6. Gabriel, C., Peyman, A., Grant, E.H.: Electrical conductivity of tissue at frequencies below 1 MHz. Phys. Med. Biol. **54**, 4863–4878 (2009)
7. Geddes, L.A. Baker, L.E.: The specific resistance of biological material – a compendium of data for the biomedical engineer and physiologist. Med. Biol. Eng. **5**, 271–293 (1967)
8. Gomez, L., Yucel, A., Hernandez-Garcia, L., Taylor, S., Michielsson, E.: Uncertainty quantification in transcranial magnetic stimulation via high dimensional model representation. IEEE Trans. Biomed. Eng. **61**, 361–372 (2015)
9. Hemingway, A., McClendon, J.F.: The high frequency resistance of human tissue. Am. J. Physiol. **102**, 56–59 (1932)
10. Hoekema, R., Wieneke, G.H., Leijten, F.S.S., van Veelen, C.W.M., van Rijen, P.C., Huiskamp, G.J.M., Ansems, J., van Huffelen, A.C.: Measurement of the conductivity of skull, temporarily removed during epilepsy surgery. Brain Topogr. **16**, 29–38 (2003)
11. Latikka, J., Kuurne, T., Eskola, H.: Conductivity of living intracranial tissues. Med. Biol. Eng. **46**, 1611–1616 (2001)
12. Starzynski, J., Sawicki, B., Wincenciak, S., Krawczyk, A., Zyss, T.: Simulation of magnetic stimulation of the brain. IEEE Trans. Magn. **38**, 1237–1240 (2002)
13. Sudret, B.: Global sensitivity analysis using polynomial chaos expansions. Reliab. Eng. Syst. Saf. **93**, 964–979 (2008)
14. Sullivan, T.J.: Introduction to Uncertainty Quantification, vol. 63. Springer, Cham (2015)
15. Tang, C., You, F., Cheng, G., Gao, D., Fu, F., Yang, G., Dong, X.: Correlation between structure and resistivity variations of the live human skull. IEEE Trans. Biomed. Eng. **55**, 2286–2292 (2008)
16. Thielscher, A., Kammer, T., Electric field properties of two commercial figure-8 coils in TMS: calculation of focality and efficiency. Clin. Neurophysiol. **115**, 1697–1708 (2004)
17. Veroy, K., Prud'homme, C., Rovas, D.V., Patera, A.T.: A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. Proceedings of the 16th AIAA Computational Fluid Dynamics Conference, vol. 3847, (2003)

18. Weise, K., Di Rienzo, L., Brauer, H., Haueisen, J., Toepfer, H.: Uncertainty analysis in transcranial magnetic stimulation using non-intrusive polynomial chaos expansion. IEEE Trans. Magn. **51**, Article: 5000408 (2015)
19. Windhoff, M., Opitz, A., Thielscher, A.: Electric field calculations in brain stimulation based on finite elements: an optimized processing pipeline for the generation and usage of accurate individual head models. Hum. Brain Mapp. **34**, 923–935 (2013)
20. Xiu, D.: Numerical Methods for Stochastic Computations: A Spectral Method Approach. Princeton University Press, Princeton (2010)

Chapter 27
Model Order Reduction of Nonlinear Eddy Current Problems Using Missing Point Estimation

Y. Paquay, O. Brüls, and C. Geuzaine

Abstract In electromagnetics, the finite element method has become the most used tool to study several applications from transformers and rotating machines in low frequencies to antennas and photonic devices in high frequencies. Unfortunately, this approach usually leads to (very) large systems of equations and is thus very computationally demanding. This contribution compares three model order reduction techniques for the solution of nonlinear low frequency electromagnetic applications (in the so-called magnetoquasistatic regime) to efficiently reduce the number of equations—leading to smaller and faster systems to solve.

27.1 Introduction

The Finite Element (FE) method has been used in numerous engineering fields to simulate various phenomena, from structural analysis to combustion modelling to electromagnetics. While its main advantage is to correctly represent dynamical and nonlinear behaviours, the spatial discretization inherent to the FE method is also its main drawback, as it usually leads to (very) large systems of (nonlinear) equations. This extensive number of equations requires a lot of computational resources, usually far too much for quasi-real time simulations.

In this paper, we propose to apply a methodology that combines the Proper Orthogonal Decomposition (POD) [19] and the Missing Point Estimation (MPE) [2] to reduce those large FE systems for nonlinear eddy current applications, e.g. for the modeling of a 3-phase power transformer, to only dozens of equations—and therefore allowing a drastic reduction in the computational time and required

Y. Paquay (✉) • C. Geuzaine
Montefiore Institute, University of Liège, Allée de la découverte B28, B-4000 Liège, Belgium
e-mail: yannick.paquay@ulg.ac.be; cgeuzaine@ulg.ac.be

O. Brüls
Department of Aerospace and Mechanical Engineering, University of Liège, Allée de la dècouverte B52, B-4000 Liège, Belgium
e-mail: o.bruls@ulg.ac.be

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_27

resources. This paper also presents a discussion on the use of the Discrete Empirical Interpolation Method (DEIM) [6] in the magnetoquasistatic case, which has already been efficiently used in the static case [7].

27.2 Eddy current problem

Let us consider a general spatial domain Ω (boundary Γ) where the nonlinear eddy current problem is to be solved in the time domain during T seconds with N_t equispaced timesteps—the corresponding time increment $\Delta t = T/N_t$. In this problem, the source is imposed directly as a current density $\mathbf{j}$ in a source domain $\Omega_j \subset \Omega$. This current density $\mathbf{j}$ generates a magnetic field $\mathbf{h}$ and a corresponding induction field $\mathbf{b} = \mu \mathbf{h}$ in Ω where μ is the permeability of the medium ($\mu = 1/\nu$ with ν the reluctivity). In general, Ω consists of linear and nonlinear magnetic subdomains, Ω^l and Ω^{nl} respectively. In Ω^l, the reluctivity is constant (e.g. $\nu = \nu_0$ with ν_0 the vacuum reluctivity) whereas in Ω^{nl} it depends on the induction field $\mathbf{b}$, i.e. $\nu = \nu(\mathbf{b})$. Parts $\Omega_c \subset \Omega$ ($\Omega_c \cap \Omega_j = \emptyset$) can be conducting with conductivity σ, in which induced currents will arise if $\mathbf{j}$ is time varying. The conductivity and the nonlinear reluctivity of a material are independent, e.g. a material can be conductive and nonlinear and would be written as Ω_c^{nl}.

The general nonlinear eddy current problem is derived from Maxwell's equations where displacement currents are neglected, and can be formulated in terms of the magnetic vector potential $\mathbf{a} \in \mathbf{H}(\mathbf{curl}, \Omega) \triangleq \left\{\mathbf{a} \in \mathbf{L}^2(\Omega); \mathbf{curl}\, \mathbf{a} \in \mathbf{L}^2(\Omega)\right\}$ such that $\mathbf{b} = \mathbf{curl}\, \mathbf{a}$ [4]:

$$\sigma \partial_t \mathbf{a} + \mathbf{curl}\ [\nu(\mathbf{curl}\, \mathbf{a})\, \mathbf{curl}\, \mathbf{a}] = \mathbf{j} \text{ in } \Omega, \tag{27.1}$$

$$\mathbf{a} \times \mathbf{n} = 0 \text{ in } \Gamma, \tag{27.2}$$

where $\mathbf{n}$ is the outer unit normal vector. Multiplying Eq. (27.1) by appropriate test functions and integrating by part over Ω leads to the following weak formulation: find $\mathbf{a}$ such that

$$\left(\sigma \partial_t \mathbf{a}, \mathbf{a}'\right)_{\Omega_c} + \left(\nu \mathbf{curl}\, \mathbf{a}, \mathbf{curl}\, \mathbf{a}'\right)_{\Omega} + \left\langle \mathbf{n} \times \nu \mathbf{curl}\, \mathbf{a}, \mathbf{a}'\right\rangle_{\Gamma} = \left(\mathbf{j}, \mathbf{a}'\right)_{\Omega_j} \tag{27.3}$$

holds for all test functions $\mathbf{a}' \in \mathbf{H}_0(\mathbf{curl}, \Omega) \triangleq \{\mathbf{a}' \in \mathbf{H}(\mathbf{curl}, \Omega);\ \mathbf{a}' \times \mathbf{n} = 0|_{\Gamma}\}$. The second term can be decomposed in the linear and nonlinear subdomains as

$$\left(\nu \mathbf{curl}\, \mathbf{a}, \mathbf{curl}\, \mathbf{a}'\right)_{\Omega} = \left(\nu_0 \mathbf{curl}\, \mathbf{a}, \mathbf{curl}\, \mathbf{a}'\right)_{\Omega^l} + \left(\tilde{\nu}(\mathbf{curl}\, \mathbf{a})\mathbf{curl}\, \mathbf{a}, \mathbf{curl}\, \mathbf{a}'\right)_{\Omega^{nl}}. \tag{27.4}$$

Applying the standard Galerkin finite element method using Whitney edge elements [9] on Eq. (27.3) leads to the spatially discretized system of differential algebraic equations [7]:

$$\mathbf{M}\dot{\mathbf{x}} + \mathbf{S}(\mathbf{x})\mathbf{x} = \mathbf{v}, \tag{27.5}$$

where $\mathbf{x}$ is the vector of unknowns of size N, $\mathbf{M}$ is the mass matrix that represents the dynamics, $\mathbf{S}$ is the magnetic stiffness matrix and $\mathbf{v}$ depends on the source current density $\mathbf{j}$.

Applying an implicit Euler scheme for the time discretisation of Eq. (27.5) leads to the discrete system of equations at time $t_k = k\Delta t$ for $k = 1, \cdots, N_t$:

$$\left[\frac{\mathbf{M}}{\Delta t} + \mathbf{S}(\mathbf{x}_k)\right]\mathbf{x}_k = \frac{\mathbf{M}}{\Delta t}\mathbf{x}_{k-1} + \mathbf{v}_k \tag{27.6}$$

with $\mathbf{x}_k = \mathbf{x}(t_k)$ and $\mathbf{v}_k = \mathbf{v}(t_k)$.

A Newton-Raphson (NR) scheme is used to linearize Eq. (27.6) at each time step. To this end, starting from an initial guess $\mathbf{x}_k^0 = \mathbf{x}_{k-1}$ and $\mathbf{x}_0^0 = \mathbf{0}$, the linear system

$$\mathbf{J}(\mathbf{x}_k^i)\delta\mathbf{x}_k^i = \mathbf{r}(\mathbf{x}_k^i) \tag{27.7}$$

is solved and the solution is updated with

$$\mathbf{x}_k^{i+1} = \mathbf{x}_k^i + \delta\mathbf{x}_k^i \tag{27.8}$$

for $i = 1, \cdots, n$ such that $\left\|\delta\mathbf{x}_k^n\right\|_2 \leq 10^{-5}$, i.e. until the increment is sufficiently small, at which point $\mathbf{x}_k$ is taken as $\mathbf{x}_k^n$. In Eq. (27.7), $\mathbf{J}(\mathbf{x}_k^i)$ is the Jacobian matrix depending on $\mathbf{x}_k^i$ and $\mathbf{r}_k^i$ is the residual given by

$$\mathbf{r}(\mathbf{x}_k^i) = \frac{\mathbf{M}}{\Delta t}\mathbf{x}_{k-1} + \mathbf{v}_k - \left[\frac{\mathbf{M}}{\Delta t} + \mathbf{S}(\mathbf{x}_k^i)\right]\mathbf{x}_k^i. \tag{27.9}$$

27.3 Model Order Reduction

The size of Eq. (27.7) equals the size N of the unknown vector $\mathbf{x}$, which can be (very) large for practical engineering simulations. This section aims at defining successful techniques to reduce the system size (and thus the CPU time required to obtain the solution). Three methods are considered: the Proper Orthogonal Decomposition (POD) [18], the Discrete Empirical Interpolation Method (DEIM) [6] and the Missing Point Estimation (MPE) [2].

27.3.1 Proper Orthogonal Decomposition

The POD is applied to reduce the system of Eq. (27.6) or the NR system from Eq. (27.7) by using a snapshot matrix $\mathbf{X}$ [19] that gathers the solutions for all time steps (called snapshots):

$$\mathbf{X} = \left[\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_{N_t} \right] \in \mathbf{R}^{N \times N_t} \tag{27.10}$$

where N_t is the number of time steps. Contrary to [7] where the calculation of the SVD is made using the covariance matrix (i.e. $\mathbf{X}^T\mathbf{X}$), we directly perform a thin singular value decomposition (SVD) on the snapshot matrix $\mathbf{X}$—maintaining the same efficiency but without computing the covariance matrix. Then

$$[\mathbf{U}_x, \Xi_x, \mathbf{V}_x] = \text{thin svd}(\mathbf{X}) \tag{27.11}$$

and the reduced basis is given by $\mathbf{U}_x$. The vector $\mathbf{x} \in \mathbf{R}^{N \times 1}$ is reduced in the basis $\mathbf{U}_x$ to a vector $\tilde{\mathbf{x}} \in \mathbf{R}^{r \times 1}$ ($r \ll N$):

$$\mathbf{x} = \mathbf{U}_x \tilde{\mathbf{x}}. \tag{27.12}$$

The reduced solution $\tilde{\mathbf{x}}$ obtained by projecting $\mathbf{x}$ onto the reduced basis $\mathbf{U}_x$ given by the application of the SVD on a snapshot matrix has been shown as the optimal (best) choice [21]. At this point, r equals N_t and in typical cases $N_t \ll N$. Nevertheless, from [21], we could also truncate the reduced basis $\mathbf{U}_x$ to its $r < N_t$ first columns in order to approximate the original snapshot matrix with a given error ε_r. Similarly, the reduced basis $\mathbf{U}_x$ could be truncated based on the Kolmogorov r-width K_r, which measures the extent to which $\mathbf{X}$ can be approximated by a r-dimensional subspace of a normed linear space [8, 13, 14]. In practice, K_r and ε_r decrease monotonically with r. Thanks to the singular values, the error ε_r is computed as:

$$\varepsilon_r = \frac{\sum_{i=r+1}^{n} \xi_i^2}{\sum_{i=1}^{n} \xi_i^2} \tag{27.13}$$

where ξ_i is the ith singular value in the diagonal of Ξ_x. The lower ε_r (e.g. $\varepsilon_r \leq 10^{-8}$), the better; graphically, this indicator measures the decay of the singular values: the fastest, the better.

By injecting Eq. (27.12) into Eq. (27.6), the reduced system is overdetermined with N equations for r unknowns:

$$\left[\frac{\mathbf{M}}{\Delta t} + \mathbf{S}(\mathbf{U}_x \tilde{\mathbf{x}}_k) \right] \mathbf{U}_x \tilde{\mathbf{x}}_k = \frac{\mathbf{M}}{\Delta t} \mathbf{U}_x \tilde{\mathbf{x}}_{k-1} + \mathbf{v}_k \tag{27.14}$$

and by applying a Galerkin projection onto the same reduced basis [21], the reduced system becomes:

$$\left[\frac{\tilde{\mathbf{M}}}{\Delta t} + \tilde{\mathbf{S}}(\mathbf{U}_x\tilde{\mathbf{x}}_k)\right]\tilde{\mathbf{x}}_k = \frac{\tilde{\mathbf{M}}}{\Delta t}\tilde{\mathbf{x}}_{k-1} + \tilde{\mathbf{v}}_k \tag{27.15}$$

where $\tilde{\mathbf{M}} = \mathbf{U}_x^T\mathbf{M}\mathbf{U}_x$ (similarly for $\tilde{\mathbf{S}}$) and $\tilde{\mathbf{v}}_k = \mathbf{U}_x^T\mathbf{v}_k$. By analogy for the NR method, Eq. (27.7) becomes

$$\tilde{\mathbf{J}}(\tilde{\mathbf{x}}_k^i)\delta\tilde{\mathbf{x}}_k^i = \tilde{\mathbf{r}}(\tilde{\mathbf{x}}_k^i) \tag{27.16}$$

with the corresponding reduced matrices

$$\tilde{\mathbf{J}}(\tilde{\mathbf{x}}_k^i) = \mathbf{U}_x^T\mathbf{J}(\mathbf{U}_x\tilde{\mathbf{x}}_k^i)\mathbf{U}_x, \tag{27.17}$$

$$\tilde{\mathbf{r}}(\tilde{\mathbf{x}}_k^i) = \frac{\tilde{\mathbf{M}}}{\Delta t}\tilde{\mathbf{x}}_{k-1} + \tilde{\mathbf{v}}_k - \left[\frac{\tilde{\mathbf{M}}}{\Delta t} + \tilde{\mathbf{S}}(\mathbf{U}_x\tilde{\mathbf{x}}_k^i)\right]\tilde{\mathbf{x}}_k^i. \tag{27.18}$$

The size of the reduced system of Eq. (27.16) is $r \ll N$ as expected but the nonlinear parts in Eqs. (27.17) and (27.18) (i.e. $\tilde{\mathbf{S}}(\mathbf{U}_x\tilde{\mathbf{x}}_k^i)$ and $\tilde{\mathbf{J}}(\mathbf{U}_x\tilde{\mathbf{x}}_k^i)$) still depend on the full order solution $\mathbf{x}_k^i = \mathbf{U}_x\tilde{\mathbf{x}}_k^i$. Therefore the evaluation of these terms still requires to expand the reduced states to the full order size solution at each nonlinear iteration.

27.3.2 Discrete Empirical Interpolation Method

The DEIM [6] (or its continuous version EIM [3]) is a nonlinear reduction technique that projects a few evaluations of a large vector (or matrix) onto a smaller mapping basis in order to reduce the computational time originally required to generate it. Let us consider the large vector $\mathbf{z}(\mathbf{p}) \in \mathbf{R}^{N\times 1}$ depending on some parameters $\mathbf{p}$ and construct the snapshot matrix $\mathbf{Z}$ as:

$$\mathbf{Z} = \left[\mathbf{z}(\mathbf{p}_1), \mathbf{z}(\mathbf{p}_2), \cdots\right]. \tag{27.19}$$

One would like to write

$$\mathbf{z}(\mathbf{p}) \simeq \mathbf{U}_z\bar{\mathbf{z}}(\mathbf{p}) \tag{27.20}$$

where $\mathbf{U}_z \in \mathbf{R}^{N\times q}$ is a mapping basis and $\bar{\mathbf{z}}(\mathbf{p}) \in \mathbf{R}^{q\times 1}$ a reduced evaluation of $\mathbf{z}(\mathbf{p})$ with $q \ll N$. The matrix $\mathbf{U}_z$ is computed as the reduced basis $\mathbf{U}_x$ in Sect. 27.3.1 by applying a thin SVD on the snapshot matrix $\mathbf{Z}$ (and can be truncated by analyzing the singular values decay in Ξ_z):

$$[\mathbf{U}_z, \Xi_z, \mathbf{V}_z] = \text{thin svd}(\mathbf{Z}) \tag{27.21}$$

The DEIM expresses $\bar{\mathbf{z}}$ from the evaluation of only q components of $\mathbf{z}$ such that

$$\bar{\mathbf{z}}(\mathbf{p}) \simeq \left(\mathbf{P}^T\mathbf{U}_z\right)^{-1}\mathbf{P}^T\mathbf{z}(\mathbf{p}) \tag{27.22}$$

with $\mathbf{P} \in \mathbf{R}^{N\times q}$ a selection matrix for the q rows of $\mathbf{z}(\mathbf{p})$ which is found by applying the DEIM algorithm [6]. Here are the main steps of this procedure:

1. From Eq. (27.20), multiplying both sides by $\mathbf{P}^T$ to select q rows of $\mathbf{z}(\mathbf{p})$ leads to:

$$\mathbf{P}^T\mathbf{z}(\mathbf{p}) \simeq \left(\mathbf{P}^T\mathbf{U}_z\right)\bar{\mathbf{z}}(\mathbf{p}). \tag{27.23}$$

2. If $\left(\mathbf{P}^T\mathbf{U}_z\right)$ is invertible, then we can deduce the expression of $\bar{\mathbf{z}}(\mathbf{p})$ (Eq. 27.22) as:

$$\bar{\mathbf{z}}(\mathbf{p}) = \left(\mathbf{P}^T\mathbf{U}_z\right)^{-1}\mathbf{P}^T\mathbf{z}(\mathbf{p}). \tag{27.24}$$

3. Finally, by injecting Eq. (27.24) into Eq. (27.20), we obtain:

$$\mathbf{z}(\mathbf{p}) \simeq \mathbf{U}_z\left(\mathbf{P}^T\mathbf{U}_z\right)^{-1}\mathbf{P}^T\mathbf{z}(\mathbf{p}). \tag{27.25}$$

4. Since a FE element only depends on its neighbours (e.g. local influence), we can restrict the computations to these q local components without generating the overall vector $\mathbf{z}(\mathbf{p})$ (similarly with rows for matrices). Equation (27.25) can therefore be written as:

$$\mathbf{z}(\mathbf{p}) \simeq \mathbf{U}_z\left(\mathbf{P}^T\mathbf{U}_z\right)^{-1}\mathbf{z}(\mathbf{P}^T\mathbf{p}). \tag{27.26}$$

In the magnetoquasistatic case from Eqs. (27.17) and (27.18), the vectors $\tilde{\mathbf{S}}(\mathbf{U}_x\tilde{\mathbf{x}}_k^i)\tilde{\mathbf{x}}_k^i$ and $\tilde{\mathbf{J}}(\mathbf{U}_x\tilde{\mathbf{x}}_k^i)\tilde{\mathbf{x}}_k^i$ perfectly match the expression of $\mathbf{z}(\mathbf{p})$. Indeed, these vectors need to evaluate $\mathbf{S}(\mathbf{x}_k^i)$ and $\mathbf{J}(\mathbf{x}_k^i)$ respectively at each nonlinear iteration. By applying the DEIM with $\mathbf{z}(\mathbf{p}_k) = \mathbf{S}(\mathbf{x}_k^n)\mathbf{x}_k^n$, these expressions read:

$$\mathbf{S}(\mathbf{x}_k^i) \simeq \mathbf{U}_z\left(\mathbf{P}^T\mathbf{U}_z\right)^{-1}\mathbf{S}(\mathbf{P}^T\mathbf{x}_k^i), \tag{27.27}$$

$$\mathbf{J}(\mathbf{x}_k^i) \simeq \underbrace{\mathbf{U}_z\left(\mathbf{P}^T\mathbf{U}_z\right)^{-1}}_{\mathbf{U}_z^*}\mathbf{J}(\mathbf{P}^T\mathbf{x}_k^i), \tag{27.28}$$

where $\mathbf{U}_z^* \in \mathbf{R}^{N\times q}$ can be computed once. By injecting Eq. (27.27) in Eq. (27.18), the reduced residual becomes:

$$\tilde{\mathbf{r}}(\tilde{\mathbf{x}}_k^i) \simeq \frac{\tilde{\mathbf{M}}}{\Delta t}\tilde{\mathbf{x}}_{k-1} + \tilde{\mathbf{v}}_k - \left[\frac{\tilde{\mathbf{M}}}{\Delta t} + \mathbf{U}_x^T\mathbf{U}_z^*\mathbf{S}(\mathbf{P}^T\mathbf{U}_x\tilde{\mathbf{x}}_k^i)\mathbf{U}_x\right]\tilde{\mathbf{x}}_k^i. \tag{27.29}$$

As will be seen later in Sect. 27.4.2, the combination of POD with DEIM lacks robustness for the considered nonlinear eddy current problem. As an alternative, we investigate below the use of the Missing Point Estimation technique [2].

27.3.3 Missing Point Estimation

The MPE approach [2] has the same goal as the DEIM: reducing the computation of all entries of a general (nonlinear) vector or matrix. While the DEIM can be used alone to approximate a vector $\mathbf{z}$ based on a small set of evaluations projected onto a reduced basis $\mathbf{U}_z$ computed from (nonlinear) snapshots $\mathbf{Z}$ of the full size vector $\mathbf{z}$, the MPE must be combined with the POD since it replaces the projection subspace $\mathbf{U}_x^T$ by another subspace defined as $\mathbf{U}_x^T\mathbf{P}\mathbf{P}^T$. As a consequence, the reduction procedure follows a Petrov-Galerkin approach with different left and right projection subspaces. Other techniques use the same philosophy, e.g. Hyper-Reduction [17] or Gappy POD [5], and differ in the determination of the reduced number of evaluations.

Let us consider the term $\tilde{\mathbf{S}}(\mathbf{x}_k^i) = \mathbf{U}_x^T\mathbf{S}(\mathbf{x}_k^i)\mathbf{U}_x$ with $\mathbf{S}(\mathbf{x}_k^i)$ that still depends on the full size order solution $\mathbf{x}_k^i$. Applying the MPE on $\mathbf{S}(\mathbf{x}_k^i)$ gives a reduced set of its rows $\bar{\mathbf{S}}(\mathbf{x}_k^i)$:

$$\bar{\mathbf{S}}(\mathbf{x}_k^i) = \mathbf{P}^T\mathbf{S}(\mathbf{x}_k^i) \tag{27.30}$$

with $\mathbf{P} \in \mathbf{R}^{N\times q}$ ($q \ll N$) a selection matrix that gathers q rows of $\mathbf{S}$ (as previously explained, only the q rows of $\mathbf{S}$ are computed, i.e. $\mathbf{S}(\mathbf{P}^T\mathbf{x}_k^i)$). Since q rows are selected in $\mathbf{S}$, only the corresponding q rows in the POD basis are useful and then kept:

$$\bar{\mathbf{U}}_x = \mathbf{P}^T\mathbf{U}_x \tag{27.31}$$

with $\bar{\mathbf{U}}_x \in \mathbf{R}^{q\times r}$ computed once (or offline). By applying the MPE on Eq. (27.16) it reads:

$$\bar{\mathbf{U}}_x^T\bar{\mathbf{J}}(\mathbf{x}_k^i)\mathbf{U}_x\delta\tilde{\mathbf{x}}_k^i = \bar{\mathbf{U}}_x^T\bar{\mathbf{r}}(\mathbf{x}_k^i) \tag{27.32}$$

with $\bar{\mathbf{J}}(\mathbf{x}_k^i) = \mathbf{P}^T\mathbf{J}(\mathbf{x}_k^i) = \mathbf{J}(\mathbf{P}^T\mathbf{x}_k^i)$ and $\bar{\mathbf{r}}(\mathbf{x}_k^i) = \mathbf{P}^T\mathbf{r}(\mathbf{x}_k^i) = \mathbf{r}(\mathbf{P}^T\mathbf{x}_k^i)$. The overall system is reduced to an r-dimensional subspace (with the application of the POD basis $\mathbf{U}_x$) but only by considering q components of the FE model (using $\mathbf{P}$) with $r, q \ll N$. Contrary to the DEIM greedy algorithm which selects the points based on the snapshots matrix $\mathbf{Z}$, the MPE greedy algorithm tends to verify

$$\bar{\mathbf{U}}_x^T\bar{\mathbf{U}}_x \approx I \tag{27.33}$$

by increasing sequentially q with the most contributing rows [2] (this procedure may be long and should be done during an offline stage). While the DEIM strongly depends on the number of snapshots to determine the reduced set of unknowns, the MPE considers the original N FE degrees of freedom. In the worst case, $q = N_t$ with the DEIM and may be too small to correctly represent the nonlinear vector/matrix or $q = N$ with the MPE and all the degrees of freedom are taken into account. The selection is based on the criteria of Eq. (27.33) and can be equivalently seen as the decay of the condition number of $\bar{\mathbf{U}}_x^T\bar{\mathbf{U}}_x$ to 1. The closest the condition number is to 1, the better the criteria is fulfilled.

27.4 Numerical Results

As a test case, we consider a 2-D nonlinear model of a 3-phase power transformer such as depicted in Fig. 27.1 [10]. The model has $N = 7300$ unknowns and is simulated at no load. The nonlinear core reluctivity is given by the Brauer law:

$$\nu(\mathbf{b}) = \gamma + \alpha \exp(\beta \mathbf{b}^2) \tag{27.34}$$

with $\gamma = 80.47$, $\alpha = 0.05$ and $\beta = 4.21$ (from core material V330-50A [10]). The (laminated) core conductivity is either chosen as zero (nonconducting) or as

$$\sigma = \frac{d^2}{12}\sigma_{\text{iron}} = 4.16 \cdot 10^{-1}\ \text{S/m} \tag{27.35}$$

where d is the thickness of the laminations (0.5 mm) and $\sigma_{\text{iron}} = 2 \cdot 10^7$ S/m [11]. A single period at 50 Hz ($T = 20$ ms) with $N_t = 20$ time steps is analyzed and the input current density is given for phase i by:

$$\mathbf{j} = (-1)^{\eta}\frac{I}{S_c}\cos(2\pi f t + \phi_i)\hat{\mathbf{e}}_z \tag{27.36}$$

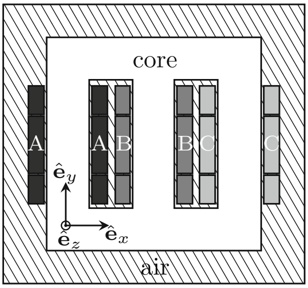


Fig. 27.1 FE model of the 3-phase transformer

Table 27.1 Subdomains of the 3-phase transformer in Fig. 27.1

Domain	Physical region	Legend in Fig. 27.1
$\Omega_c^{nl} \setminus \Omega_j$	Core (if $\sigma \neq 0$)	White
Ω_{cc}^{l}	Air	Lined
Ω_{cc}^{nl}	Core (if $\sigma = 0$)	White
Ω_j	Windings	Filled

Phase	Phase delay ϕ [rad]	Legend in Fig. 27.1
A	0	Black
B	$4\pi/3$	Gray
C	$2\pi/3$	Light gray

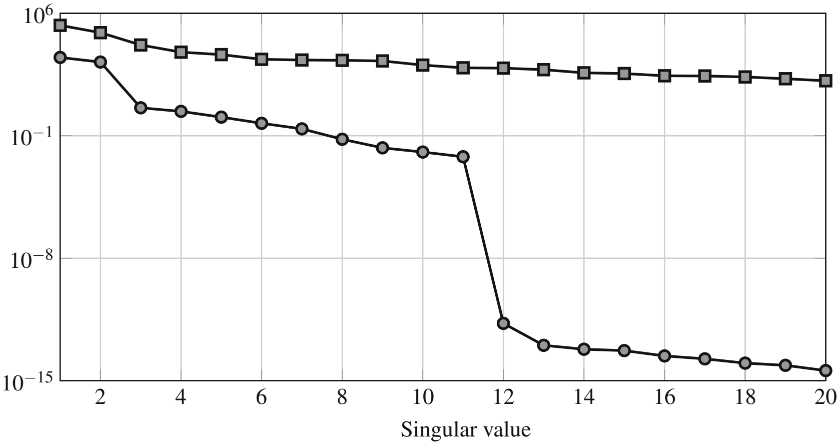


Fig. 27.2 Singular values of snapshot matrix **X**, i.e. Ξ_x (*circled with line*) and snapshot matrix **Z**, i.e. Ξ_z (*squared with line*) with $I = 0.3$ A & $\sigma = 4.16 \cdot 10^{-1}$ S/m

where $\eta = 0$ (resp. $\eta = 1$) for left (resp. right) part of the coil (representing the direction of the current), $I \in [0.1, 0.3]$ is the input peak current ($I = 0.1$ A induces linear magnetic behaviour whereas $I = 0.3$ A causes the core to saturate, see Fig. 27.9), S_c the coil surface, ϕ_i the phase delay of phase i and $\hat{\mathbf{e}}_z$ the unit vector along the z-axis. A full-order time domain simulation of 20 ms takes around 1 min to compute. The subdomains description is given in Table 27.1.

27.4.1 Proper Orthogonal Decomposition

First, in the reduction process, one must verify that the problem can be mapped onto a smaller r-dimensional subspace. Since the singular values of the snapshot matrix **X** quickly decay (circled with line in Fig. 27.2), the POD can indeed be used to reduce the system while preserving a small error. In the following tests, the POD basis is truncated after the 11th singular value to fulfill $\varepsilon_r = 10^{-15}$ with $r = 11$. As explained in [16] for an inductor-core system, varying the input current does not always require to recompute the POD basis if the model is well trained with an appropriate current to correctly capture the nonlinear behaviour. By defining the reduced snapshot matrix $\tilde{\mathbf{X}}$ as:

$$\tilde{\mathbf{X}} = \left[\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2, \cdots, \tilde{\mathbf{x}}_{N_t}\right] \in \mathbf{R}^{r \times N_t}, \tag{27.37}$$

and using Eq. (27.12), we define the relative error of method i as:

$$\gamma_i = \frac{\left\| \mathbf{X} - \mathbf{U}_x \tilde{\mathbf{X}}_i \right\|_2}{\|\mathbf{X}\|_2}, \tag{27.38}$$

where $\tilde{\mathbf{X}}_i$ collects the reduction states obtained by reduction technique i. In this 3-phase transformer, a single POD basis can achieve a small relative error for all input currents—below 5% from an engineering point of view (dashed line in Fig. 27.3). Unfortunately, the perfect choice of that basis is very sensitive. Secondly, changing the conductivity value requires another POD basis due to a change in the eddy current distribution—similar to a change in frequency [16] (straight line in Fig. 27.4). For practical applications though, contrary to the input current, once the transformer is built, the conductivity is fixed and is no longer a parameter. If a local

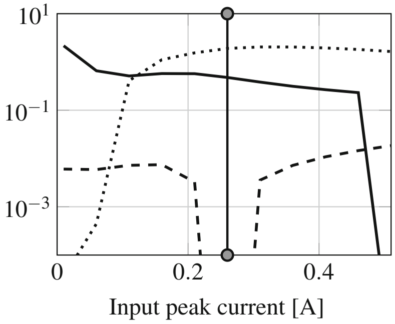


Fig. 27.3 Relative error γ_{POD} according to input peak current (basis generated with input current $I = 0.01$ A *dotted line*, 0.25 A *dashed line*, 0.5 A *straight line*). *Circled with line* represents the transition between linear and nonlinear regimes

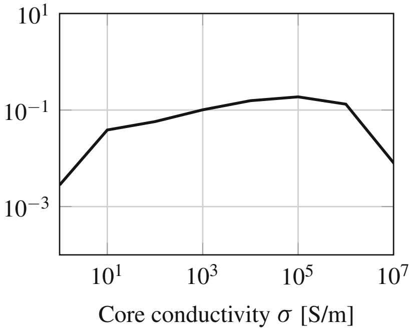


Fig. 27.4 Relative error γ_{POD} according to core conductivity (basis generated with $\sigma = 1$ S/m)

basis cannot correctly represent all the dynamics in a single low dimensional space, one would build a global basis by gathering all the snapshots of all parameter values [7] or interpolate between the local reduced bases in the parameter space [1, 16].

The POD reduces the magnetoquasistatic case to 11 equations leading to a theoretical speedup between 663 and 440,413 (the computational time plummets to between 90 and 0.1 ms) depending on the linear solver [12]. But the nonlinear terms still need the full order size: much of the computational gain is lost there, and the application of the POD alone is thus not sufficient.

27.4.2 Discrete Empirical Interpolation Method

In [7], the POD-DEIM has been used to efficiently reduce a static ($\sigma = 0$ S/m) 3D 3-phase transformer with an error lower than 0.1% by using 55 DEIM components (representing edges in the model). Once the core conductivity is no longer zero, however, the stability of the DEIM suffers and becomes more and more dependent on a priori independent parameters such as the number of time steps N_t or the conductivity σ [15, 20]. An alternative to the original DEIM algorithm is proposed in [22], i.e. DIME, but also presents the same issues in this eddy current problem (3-phase transformer, $I = 0.3$ A and $\sigma = 4.16 \cdot 10^{-1}$ S/m). This lack of robustness is illustrated in Figs. 27.5 and 27.6 where changing the number of time steps highly impacts the relative error obtained with the POD-DEIM squared with line and POD-DIME squared with dashed line techniques whereas the POD-MPE straight line approach keeps a quasi constant relative error. In Fig. 27.5, the DEIM/DIME reduced size q is obtained by truncating the nonlinear basis $\mathbf{U}_z$ through the SVD to maintain $\varepsilon \leq 10^{-15}$. In practice, the 25 first modes are significant (i.e. when $N_t \geq 25$). In Fig. 27.6, no truncation is performed on $\mathbf{U}_z$ and $q = N_t$. The results remain unchanged for $N_t > 90$. In both figures, the MPE reduced size is kept constant as the influence of the MPE reduced size on the error is analysed in Fig. 27.8 in the following section. By looking at the singular values of the nonlinear

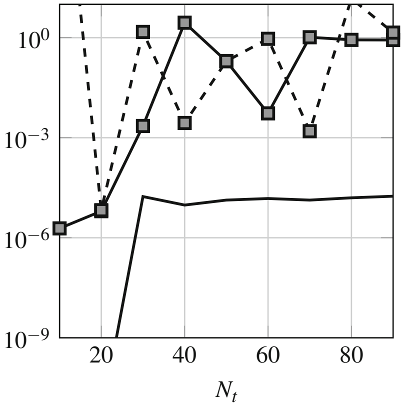


Fig. 27.5 Relative errors $\gamma_{\text{POD-DEIM}}$ (*squared with line*), $\gamma_{\text{POD-DIME}}$ (*squared with dashed line*) with $q = \min(25, N_t)$ and $\gamma_{\text{POD-MPE}}$ (*straight line*)

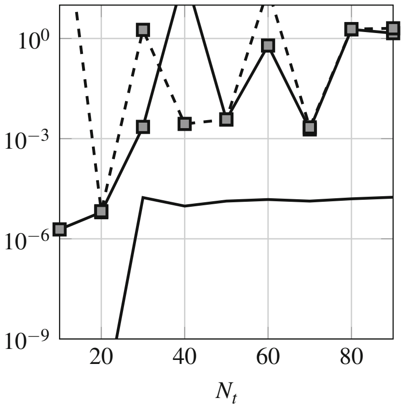


Fig. 27.6 Relative errors $\gamma_{\text{POD-DEIM}}$ (*squared with line*), $\gamma_{\text{POD-DIME}}$ (*squared with dashed line*) with $q = N_t$ and $\gamma_{\text{POD-MPE}}$ (*straight line*)

contribution, one can find a decay similar to the one depicted by squared with line in Fig. 27.2 where a small projection error is not achievable with a low number of modes (or similarly the Kolmogorov q-width is too large) to effectively represent the nonlinear term onto a q-dimensional subspace with $q \ll N$. Consequently, the POD-DEIM (and POD-DIME) can not correctly be used to reduce the computation of the nonlinear terms.

27.4.3 Missing Point Estimation

The application of the MPE consists in the determination of the q rows to keep in Eq. (27.16) to obtain Eq. (27.32). Contrary to the DEIM, it can be seen in Fig. 27.5 that the number of time steps N_t does not significantly influence the reduction. Similar results were obtained by varying the number of simulated periods T or the conductivity.

The POD-MPE is applied to the same 3-phase power transformer as the POD-DEIM (Fig. 27.1), using both the small ($I = 0.1$ A) and the large ($I = 0.3$ A) current values and either a zero ($\sigma = 0$ S/m) or nonzero ($\sigma = 4.16 \cdot 10^{-1}$ S/m) conductivity for the core. The condition number of $\bar{\mathbf{U}}_x^T \bar{\mathbf{U}}_x$, for both extreme cases, decays very fast to 1 (see Fig. 27.7). However, the relative error $\gamma_{\text{POD-MPE}}$ with respect to the reduction ratio seems to be independent of this criteria (see Fig. 27.8) where q ranges from 50 to 350 (depending on the configuration) for a relative error below 0.1%. This important reduction allows a high gain in the computational time and resources but a better criterion should be investigated. By applying the POD-MPE, the assembly of the nonlinear terms is limited to $q \in [50, \cdots, 350]$ rows instead of $N = 7300$ and the projection of them onto the reduced basis $\mathbf{U}_x$ at each nonlinear iteration is also computed faster compared to the original matrix products. The reduction ratios are comprised between 99% and 95% allowing a computational time from 0.6 to 3 s, still limited by the assembly time compared to the resolution time of 90 ms (obtained with the use of the POD allowing a drastic reduction to

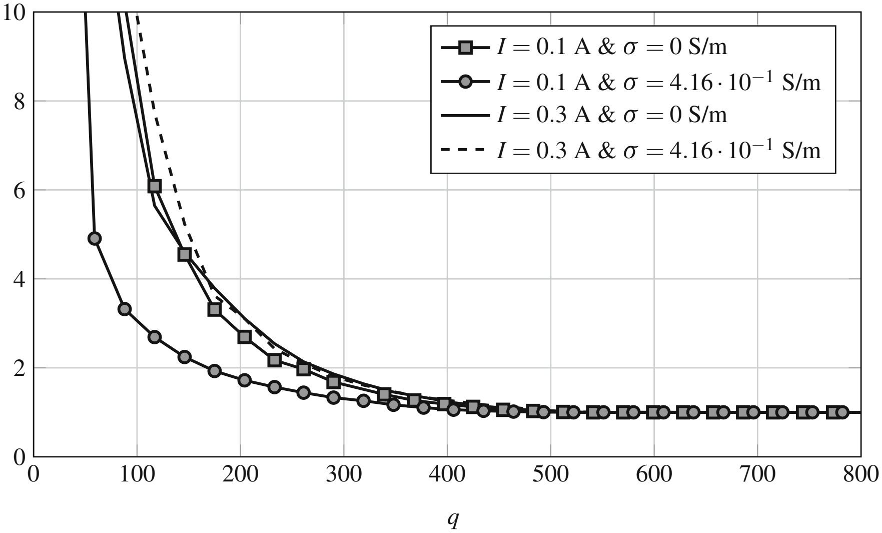


Fig. 27.7 Condition number of $\bar{\mathbf{U}}_x^T \bar{\mathbf{U}}_x$

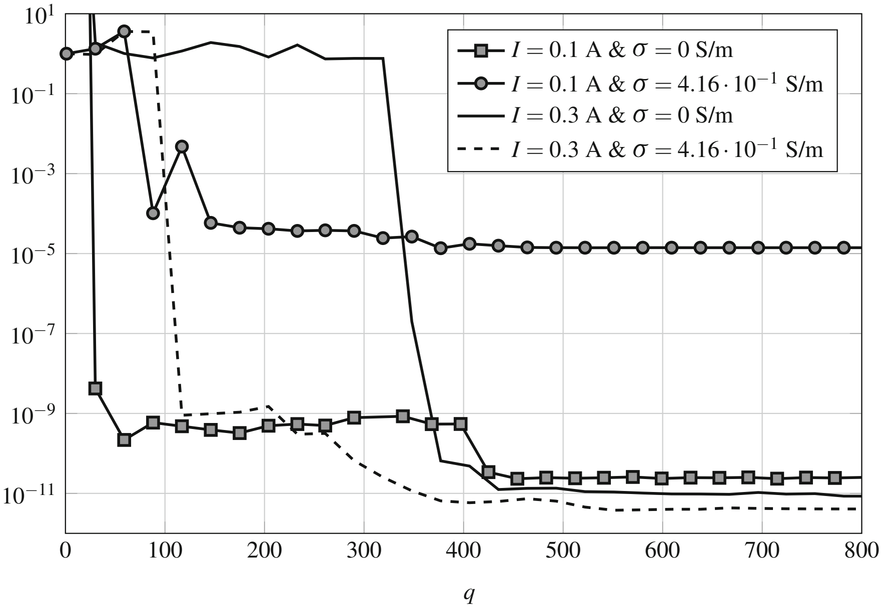


Fig. 27.8 Relative error $\gamma_{\text{POD-MPE}}$

$r = 11$). The induction field $\mathbf{b}$ for the different setups, i.e. $I = 0.1$ A (top)—$I = 0.3$ A (middle) and $\sigma = 0$ S/m (left)—$\sigma = 4.16 \cdot 10^{-1}$ S/m (right), and the MPE selected points (bottom) are shown in Fig. 27.9.

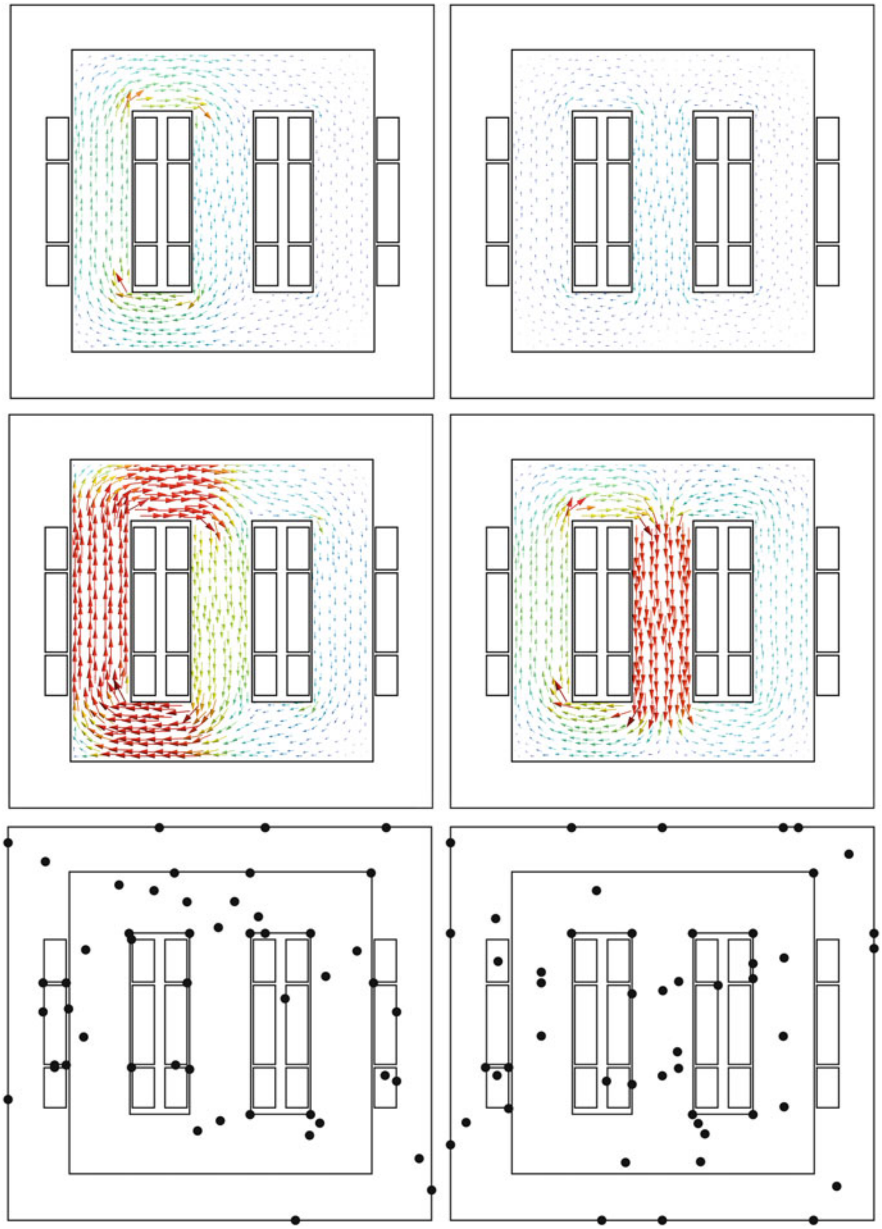


Fig. 27.9 20th time step of induction field **b** for $I = 0.1$ A (*top*) - $I = 0.3$ A (*middle*) and $\sigma = 0$ S/m (*left*) - $\sigma = 4.16 \cdot 10^{-1}$ S/m (*right*). Legend: linear scale from 0 T (*blue-small arrows*) to 1.52 T (*red-large arrows*) . 50 first MPE points (*bottom*)

27.5 Conclusion

In this paper, we investigated a combined approach of the Proper Orthogonal Decomposition and the Missing Point Estimation to efficiently and drastically reduce both nonlinear static and eddy current models of a 3-phase power transformer. The reduction ratios, comprised between 99% and 95% for the assembly and around 99.9% for the resolution, allow a reduced computational time of 0.6–5 s compared to the original finite element model resolution time of about 60 s. However, further work should investigate a better suited criterion on the a priori reduced size to ensure a sufficiently small error.

Acknowledgements This work was funded in part by the Belgian Science Policy under grant IAP P7-02 (Multiscale Modelling of Electrical Energy Systems) and by the F.R.S.—FNRS (Belgium).

References

1. Amsallem, D.: Interpolation on manifolds of CFD-based fluid and finite element-based structural reduced-order models for on-line aeroelastic predictions. Ph.D. dissertation, Stanford, USA (2010)
2. Astrid, P., et al.: Missing point estimation in models described by proper orthogonal decomposition. IEEE Trans. Autom. Control, **53**(10), 2237–2251 (2008)
3. Barrault, M., et al.: An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations. C. R. Math. **339**(9), 667–672 (2004)
4. Bossavit, A.: Computational Electromagnetism: Variational Formulations, Complementarity, Edge Elements. Academic Press, San Diego (1998)
5. Bui-Thanh, T., Damodaran, M., Willcox, K.E. : Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. AIAA J. **42**, 1505–1516 (2004)
6. Chaturantabut, S., Sorensen, D.C.: Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. **32**(5), 2737–2764 (2010)
7. Clenet, S., et al.: Model order reduction of non-linear magnetostatic problems based on POD and DEI methods. IEEE Trans. Magn. **50**, 33–36 (2014)
8. Cohen, A., DeVore, R.: Kolmogorov widths under holomorphic mappings. IMA J. Numer. Anal. (2015)
9. Dular, P., et al.: A general environment for the treatment of discrete problems and its application to the finite element method. IEEE Trans. Magn. **34**(5), 3395–3398 (1998)
10. Gyselinck, J.: Twee-Dimensionale Dynamische Eindige-Elementenmodellering van Statische en Roterende Elektromagnetische Energieomzetters. PhD thesis (1999)
11. Gyselinck, J., et al.: Calculation of eddy currents and associated losses in electrical steel laminations. IEEE Trans. Magn. **35**(3), 1191–1194 (1999)
12. Hiptmair, R., Xu, J.-C.: Nodal auxiliary space preconditioning for edge elements. In: 10th International Symposium on Electric and Magnetic Fields, France (2015)
13. Kolmogoroff, A.: Über die beste Annäaherung von Funktionen einer gegebenen Funktionenklasse. Ann. Math. Second Ser. **37**, 107–110 (1936)
14. Maday, Y., Patera, A., Turinici, G.: A priori convergence theory for reduced-basis approximations of single-parameter elliptic partial differential equations. J. Sci. Comput. **17**, 437–446 (2002)
15. Montier, L., et al: Robust model order reduction of a nonlinear electrical machine at start-up through reduction error estimation. In: 10th International Symposium on Electric and Magnetic Fields, France (2015)

16. Paquay, Y., Brüls, O., Geuzaine, C.: Nonlinear interpolation on manifold of reduced order models in magnetodynamic problems. IEEE Trans. Magn. **52**(3), 1–4 (2016)
17. Ryckelynck, D.: A priori hyperreduction method: an adaptive approach. J. Comput. Phys. **202**(1), 346–366 (2005)
18. Schilders, W., et al.: Model Order Reduction: Theory, Research Aspects and Applications, vol. 13. Springer, Berlin (2008)
19. Sirovich, L.: Turbulence and the dynamics of coherent structures. Part I: Coherent structures. Q. Appl. Math. **45**(3), 561–571 (1987)
20. Sorensen, D. Private discussions (2015)
21. Volkwein, S.: Proper orthogonal decomposition and singular value decomposition. Universität Graz/Technische Universität Graz. SFB F003-Optimierung und Kontrolle (1999)
22. Zlatko, D., Gugercin, S.: A New Selection Operator for the Discrete Empirical Interpolation Method–improved a priori error bound and extensions. SIAM J. Sci. Comput. **38**(5), A631–A648 (2016)

Chapter 28
On Efficient Approaches for Solving a Cake Filtration Model Under Parameter Variation

S. Osterroth, O. Iliev, and R. Pinnau

Abstract In this work, we are considering a mathematical model for an industrial cake filtration process. The model is of moving boundary type and involves a set of parameters, which vary in a given range. We are interested in the case when the model has to be solved for thousands of different parameter values, and therefore model order reduction (MOR) is desirable, so that from full order solutions with one or several sets of parameters we derive a reduced model, which is used further to perform the simulations with new parameters. We study and compare the performance of several MOR techniques known from the literature. We start with standard MOR based on proper orthogonal decomposition (POD) and consider also several more advanced techniques based on combination of MOR and reduced basis techniques, including approaches relying on computation of sensitivities. The transformation from a moving to a fixed domain introduces time varying coefficients into the equations, which makes it reasonable to use an offline/online decomposition. Several test cases involving different simulation time horizons and short time training are considered. Numerical tests show that the discussed methods can approximate the full model solution accurately and work efficiently for new parameters belonging to a given parameter range.

28.1 Introduction

Filtration and separation processes are found in an abundance of everyday applications, in buildings, vehicles, and vacuum cleaners, to name only a few and these processes are essential for ensuring a high quality of life. Thorough mathematical modeling and fast and accurate computer simulations advance the prediction of

S. Osterroth (✉) • O. Iliev
Fraunhofer ITWM, Fraunhofer-Platz 1, 67663 Kaiserslautern, Germany
e-mail: sebastian.osterroth@itwm.fraunhofer.de; iliev@itwm.fraunhofer.de

R. Pinnau
University of Kaiserslautern, Gottlieb-Daimler Str. 48, 67663 Kaiserslautern, Germany
e-mail: pinnau@mathematik.uni-kl.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_28

filtration efficiency and reduce design and manufacturing costs for more advanced filters.

In this paper, we focus on solid/liquid separation, specifically, the separation of particles from a fluid such as oil or air. Here two filtration modes are mainly used: cross-flow and dead-end filtration. In cross-flow filtration only part of the dirty fluid passes through the filtering medium. This mode is characterized by one inlet with dirty fluid and two outlets, one with cleaned fluid and one with the rest of the dirty fluid. In dead-end filtration, in which the fluid is pushed through a porous filtering media that separates dirty inflow from clean (or at least cleaner) outflow, all (or most of) the particles are deposited inside or on the surface of the medium and are thus removed from the dirty fluid. Here the filtration, which targets at capturing particles strictly within the filtering medium is called depth filtration, while deposition of particles on the surface of the filtering medium leading to a growth of so-called cake there, is known as cake filtration.

Most previous literature has focused on either depth filtration ([11] and references therein), or pure cake filtration ([26] and references therein). However, especially in the case of polydisperse particles, one often observes a simultaneous depth and cake filtration, and this process is indeed the subject of our study. A major challenge of any filtration process is to balance two key factors: the capturing efficiency and the flow rate—pressure drop ratio [11]. Whereas the first factor relates to the purity of the filtered fluid and the size of the penetrating particles, the second is primarily a measure of energy efficiency. In general, one seeks high filtration efficiency at low energy cost, but the two criteria are contradictory and as mentioned above, a balance is needed. In addition to these two key factors, a third important criterion is the dirt storage capacity, i.e., the amount of dirt that can be captured before replacing a filter. This criterion directly correlates with the lifetime of a filter. Because the three criteria impose contradictory requirements, there is no simple answer for selecting the filtering media and the operation conditions. For example, selecting a filter medium with smaller pores improves the filtration efficiency, but reduces the energy efficiency and shortens the lifetime. Therefore detailed studies are needed to make these decisions. Even for the two commonly considered cases, pure depth filtration (all captured particles deposited strictly within the filtering medium) and pure cake filtration (all particles captured and deposited on the upstream surface of the porous media), these studies are not trivial. For the combined depth and cake filtration system considered here, they are even more challenging. A model for this system and first results are described in [10]. Note that the cake described by this model is an incompressible one.

Generally, one has to differ between the microscopic description, where one models single particles and a resolved pore structure [16, 26] and the macroscopic description, where the particles are modeled as a dissolved continuum [10, 13, 25], i.e., a concentration. Here we consider the latter case, i.e., we deal with an effective porous medium and describe averaged quantities. Navier-Stokes-Brinkman equations are a standard model for describing dead-end filtration related flows through plain and porous media, we refer to [11] for more details on this topic. For the needs of this article it is just important to know that the velocity is computed in advance and it is taken as input parameter for the equations describing the transport and capturing of (concentration of) particles.

The combined depth and cake filtration is based on the equations for depth filtration [9]. The dissolved concentration in the porous medium and in the cake is described with a convection-diffusion-reaction (CDR) equation. The concentration of particles deposited inside the medium and inside the cake is described with a kinetic expression and these two equations are coupled with an evolution equation (depending on flow, filtering medium, cake and particles properties) accounting for the growth of the cake [10]. This set of equations appears for every different particle size leading to a large number of unknowns in the case of polydisperse particles. For dead-end filtration it is a reasonable approximation to assume that one spatial direction (perpendicular to the filtering medium) is considered in the CDR. However, one has to account for the variation of velocity. Overall we end up with a large-scale moving boundary value problem.

The above model exhibits a set of parameters, which are subject to parameter variation. We perform a sensitivity analysis to investigate the change of the solution with respect to a parameter. This gives us further insight in the behavior of the system. Our goal is to simulate the system for a large set of parameters. Thus we use model reduction to decrease the computational time needed to solve the system. The system is solved for a set of reference parameters and the corresponding solutions are used to derive a reduced-order model. This reduced model is applied to other parameter configurations and test cases. In our case, we modify the reduced model depending on the actual parameter values with the help of information computed from the sensitivity analysis. We use extrapolation, expansion, and interpolation [2, 7, 12]. Other authors also suggest using the sensitivity information directly in the basis determination procedure [4, 5, 23]. An important ingredient for the method is the offline/online decomposition procedure [20], which is essential to achieve a computational speed up.

The sequel of the article is organized as follows. In Sect. 28.2 the problem is described mathematically and the transformation from a moving to a fixed domain is carried out. In Sect. 28.3 the model reduction framework is described. This is followed by the description of a 1D problem describing a standard filtration test case. Finally, we present numerical results in Sect. 28.5 for three test cases demonstrating the feasibility of our approach. Concluding remarks are given in Sect. 28.6.

28.2 Model Problem

The process of cake filtration is depicted schematically in Fig. 28.1. We are considering a fluid flow from right to left with constant inflow rate.

Consider a domain Ω, which is split into two parts: the free liquid part $\Omega_f(t)$ and the filtering (porous) part $\Omega_p(t)$, i.e., $\Omega = \Omega_f(t) \cup \Omega_p(t)$. The subdomains depend on time, since the interface between them is moving while the cake grows,

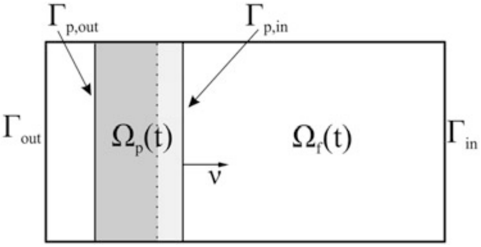


Fig. 28.1 Computational domain Ω: in *dark gray* filter medium, in *light gray* filter cake with surface normal direction $\boldsymbol{\nu}$. Flow from right (inflow boundary Γ_{in}) to left (outflow boundary Γ_{out})

whereas the overall domain Ω is fixed. The inflow boundary is denoted by Γ_{in} and the outflow boundary by Γ_{out}.

The governing equations are given below, see also [10]. Here C denotes the concentration of dissolved dirt and M is the concentration of deposited dirt inside the medium/cake (due to depth filtration).

$$\partial_t C + \mathbf{u} \cdot \nabla C - D \Delta C = \begin{cases} 0, & \mathbf{x} \in \Omega_f(t) \\ -\partial_t M, & \mathbf{x} \in \Omega_p(t) \end{cases} \tag{28.1}$$

$$\partial_t M = \alpha |u_N| C, \quad \mathbf{x} \in \Omega_p(t) \tag{28.2}$$

Equation (28.1) is describing the transport of dirt towards the medium and has an additional reaction term in the porous part of the domain, i.e., here some of the particles are deposited. Equation (28.2) describes the rate of deposition (so called adsorption rate) only in the porous part. For the moment we assume that the adsorption rates coincide in the medium and the cake. The parameters are the macroscopic fluid velocity $\mathbf{u}$, the diffusivity D and the adsorption rate α. The velocity component in normal direction $\boldsymbol{\nu}$ to the cake (or the porous medium) surface is denoted by u_N. Since the velocity is precomputed the value is known. In most of the cases also the flow in the porous medium is normal. This allows us to use the same normal velocity inside the medium. Assuming that the concentration C is constant in the liquid part of the domain, we consider only Ω_p instead of Ω. For that reason we impose the following boundary conditions:

$$C = (1 - \lambda) C_{in}, \quad \mathbf{x} \in \Gamma_{p,in} \tag{28.3}$$

$$\partial_{\boldsymbol{\nu}} C = 0, \quad \mathbf{x} \in \Gamma_{p,out}, \tag{28.4}$$

where C_{in} is a given constant inflow concentration and $\boldsymbol{\nu}$ is the direction normal to the outflow. Here λ denotes the so called cake growth factor which describes the growth of the cake thickness L as

$$\frac{\partial L}{\partial t} = \lambda \frac{C_{in} |u_N|}{\rho_s (1 - \phi_{cake})}. \tag{28.5}$$

The particle density is given by ρ_s and a prescribed cake porosity by ϕ_{cake}. The initial conditions are given as

$$C(\mathbf{x},0) = C_0(\mathbf{x}), \quad M(\mathbf{x},0) = 0, \quad L(0) = L_0, \tag{28.6}$$

with given functions C_0 and L_0.

Note that in the above model the concentration M is decoupled from C. In more involved models, e.g., a nonlinear model $\partial_t M = \alpha|u_N|(1 + M/M_0)C$, the two quantities can be coupled [9, 11]. Secondly the modeling of the whole filtration loop requires additional information. As mentioned previously, the Navier-Stokes-Brinkman equations are used to compute the flow field and the pressure drop. Here the Brinkman term includes the so-called permeability, which measures the resistance of the filter medium and the cake to the flow. Due to the growth of the cake and the loading with particles this resistance increases. To compute the change in permeability due to loading the concentration of deposited dirt M is needed (see, e.g. [11]).

Equation (28.5) is independent of the solution C and therefore the cake thickness can be precomputed. A generalization of the moving to a free boundary problem is in progress and will be presented in a forthcoming article. There, the Dirichlet condition will be replaced by a (reactive) Robin condition.

As mentioned above, we restrict our computations to the porous part of the domain. Thus this part is not fixed any longer, but evolves in time. Therefore a discretization has to be adjusted in every time step. To avoid this (also with the hidden agenda of model reduction), we transform the domain $\Omega_p(t)$ to a fixed domain Ω_{ref} independent of t. Let us define a mapping $\Psi : [0,T] \times \Omega_p(t) \to \Omega_{ref}$. Introduce a new coordinate $\mathbf{y} = \Psi(t,\mathbf{x})$ and the transformed variables

$$\bar{C}(t,\mathbf{y}) = \bar{C}(t,\Psi(t,\mathbf{x})) = C(t,\mathbf{x}), \quad \bar{M}(t,\mathbf{y}) = \bar{M}(t,\Psi(t,\mathbf{x})) = M(t,\mathbf{x}). \tag{28.7}$$

Transforming the differential operators by using the chain rule leads to

$$\mathrm{d}_t C(t,\mathbf{x}) = \partial_t \bar{C} + \sum_j \frac{\partial \bar{C}}{\partial y_j}\frac{\partial y_j}{\partial t} = \partial_t \bar{C} + \nabla \bar{C} \cdot \partial_t \mathbf{y} \tag{28.8}$$

$$\mathbf{u} \cdot \nabla C = \sum_i u_i \frac{\partial C}{\partial x_i} = \sum_i u_i \sum_j \frac{\partial \bar{C}}{\partial y_j}\frac{\partial \Psi(y_j)}{\partial x_i} = \mathbf{u} \cdot \left(D\Psi^T \nabla \bar{C}\right) \tag{28.9}$$

$$\Delta C = \nabla^T \nabla C = \left(D\Psi^T \nabla\right)^T D\Psi^T \nabla \bar{C} = \nabla^T D\Psi D\Psi^T \nabla \bar{C}, \tag{28.10}$$

where $D\Psi$ is the Jacobian matrix. Then the governing equations (28.1) and (28.2) can be rewritten in the transformed, time-independent domain Ω_{ref} as

$$\partial_t \bar{C} + (\partial_t \mathbf{y} + D\Psi \mathbf{u}) \cdot \nabla \bar{C} - D\nabla^T D\Psi D\Psi^T \nabla \bar{C} = -\partial_t \bar{M} - \partial_t \mathbf{y} \cdot \nabla \bar{M} \tag{28.11}$$

$$\partial_t \bar{M} + \partial_t \mathbf{y} \cdot \nabla \bar{M} = \alpha|u_N|\bar{C}. \tag{28.12}$$

Note that now the coefficients are depending on time. The boundary and initial conditions are transformed analogously.

28.3 Reduced Basis Method

Consider a PDE of the form

$$\partial_t w + A_1(\boldsymbol{\mu}) \cdot \nabla w - A_2(\mu)\Delta w = R(\mu)w, \tag{28.13}$$

where $\boldsymbol{\mu} \in \mathscr{D} \subset \mathbb{R}^p$ is a parameter vector from a set $\mathscr{D}$ of admissible parameter values. Here A_1, A_2 and R denote parameter dependent prefactors. Define the solution set $X_{\mathscr{D}} = \{w(\mu) : \mu \in \mathscr{D}\}$. The goal of the reduced basis method is to find a low dimensional subspace $X_{RB} \subset X_{\mathscr{D}}$ approximating the solution in a proper manner. The space X_{RB} is built from solution trajectories $w(\mu_i)$ for $i = 1, \ldots, N_s$, where N_s is the number of different parameter choices. Thus the span is given as $X_{RB} = span\left(\{w(\mu_1), \ldots, w(\mu_{N_s})\}\right)$, where the solution is included for all time instances. Such a reduced basis space is called a Lagrangian reduced basis [21]. The chosen values are mostly referred to as snapshots. In general they are not orthogonal. Therefore we use the technique proper orthogonal decomposition (POD) to extract an orthonormal basis from the snapshots. The idea of POD is to approximate the given snapshots in an optimal least-squares sense [14, 19].

We shortly summarize the most important features of POD. For more details we refer to [19, 21, 27]. Consider the minimization problem

$$\underset{\phi_1,\ldots,\phi_\ell}{\operatorname{argmin}} \mathscr{J}(\phi_1, \ldots, \phi_\ell) = \sum_{i=1}^{N_s} \left\| w(\mu_i) - \sum_{j=1}^{\ell} \langle w(\mu_i), \phi_j \rangle_W \phi_j \right\|_W^2 \tag{28.14a}$$

$$\text{s.t.} \quad \langle \phi_i, \phi_j \rangle_W = \delta_{ij}, \quad i, j = 1, \ldots, \ell, \tag{28.14b}$$

where W is a Hilbert space and $\langle \cdot, \cdot \rangle_W$ is the corresponding inner product. The solution ϕ_i, $i = 1, \ldots, \ell$, of the above problem can be obtained by solving an eigenvalue/eigenvector problem [27]. It is called POD basis or POD modes. The number ℓ has to be chosen a priori and different methods for choosing the value are available, for example energy estimates [19] or the Greedy algorithm [21].

Having a basis at hand, one can use a Galerkin projection to obtain the reduced model. Define the approximation w_ℓ of w by

$$w_\ell = \sum_{i=1}^{\ell} \eta_i(t)\phi_i. \tag{28.15}$$

Inserting into Eq. (28.13), multiplying from the right by $\phi_j, j = 1, \dots, \ell$, we end up with

$$\partial_t \eta_i + \underbrace{\sum_{i=1}^{\ell} \phi_j A_1(\mu) \cdot \nabla \phi_i \, \eta_i(t)}_{=\tilde{a}^1_{.j}(\mu)} - \underbrace{\sum_{i=1}^{\ell} \phi_j A_2(\mu) \Delta \phi_i \, \eta_i(t)}_{\tilde{a}^2_{.j}(\mu)} = \underbrace{\sum_{i=1}^{\ell} \phi_j R(\mu) \phi_i \, \eta_i(t)}_{\tilde{r}_{.j}(\mu)}. \tag{28.16}$$

This yields the reduced system

$$\partial_t \boldsymbol{\eta} + \tilde{A}_1(\mu)\boldsymbol{\eta} - \tilde{A}_2(\mu)\boldsymbol{\eta} = \tilde{R}(\mu)\boldsymbol{\eta}, \tag{28.17}$$

where $\boldsymbol{\eta} = [\eta_1, \dots, \eta_\ell]^T$ and matrices $\tilde{A}_1(\mu)$, $\tilde{A}_2(\mu)$ and $\tilde{R}(\mu)$ defined from the above entries. Let us further assume that the prefactors are affine parametric dependent

$$A_1(\mu) = \sum_{i=1}^{q_1} \theta^1_i(\mu) A^i_1, \quad A_2(\mu) = \sum_{i=1}^{q_2} \theta^2_i(\mu) A^i_2, \quad R(\mu) = \sum_{i=1}^{q_r} \theta^r_i(\mu) R^i, \tag{28.18}$$

where θ^1_i, θ^2_i and θ^r_i are scalar functions for all $i = 1, \dots, \ell$ [21]. This property allows a further simplification of (28.17) and the splitting of the computation process in an offline and an online stage [20].

The snapshots are determined from the trajectories for a given set of reference parameters $\mathscr{D}_{ref} \subset \mathscr{D}$. Consequently the derived basis yields the best approximation for this parameter setting. For $\mu \in \mathscr{D} \setminus \mathscr{D}_{ref}$ it is a priori not clear if the derived basis yields a good approximation. Thus for improving the robustness of the basis, sensitivity information are included. This can be done by either including the sensitivities of the trajectory with respect to the model parameters [5, 23] or by considering the sensitivities of the POD basis functions itself [7, 12]. From a practical point of view this investigation is very important, since in design, control or optimization one needs to perform several simulations [8]. Therefore the sensitivities of the POD basis are computed as in [7] and the methods extrapolation and expansion are used. In extrapolation a first order Taylor approximation of the POD basis around a given reference parameter μ^0 is used, i.e.,

$$\phi(\mu) = \phi(\mu^0) + \sum_{i=1}^{p} (\mu_i - \mu^0_i) \frac{\partial \phi}{\partial \mu_i}(\mu^0). \tag{28.19}$$

Note that the new basis is not guaranteed to be orthonormal. Therefore an additional re-orthonormalization might be needed. For expansion the approximation subspace X_{RB} is increased by adding the sensitivities as additional basis functions. Since for p different parameters sensitivities are added the size of the basis is $(p + 1)\ell$. Here one also has to take care of the basis properties. Another method for adjusting the basis function with respect to parameter variation is interpolation. Amsallem and Farhat introduced in [2] a technique based on a mapping to the tangent space, doing interpolation there and mapping back. This algorithm guarantees the basis properties.

28.4 The 1D Problem

For testing we simplify the model to a standard filtration test case. We assume that a flat filtering medium is used and that the flow rate is constant. Assuming further that the flow is unidirectional from one side allows to reduce the model to 1D. Then Eqs. (28.1) and (28.2) read

$$\partial_t C + u\partial_x C - D\partial^2_{xx} C = -\partial_t M, \quad x \in \Omega_p(t), \tag{28.20}$$

$$\partial_t M = \alpha |u| C, \quad x \in \Omega_p(t), \tag{28.21}$$

where $\Omega_p(t) = [0, L(t)] \times A$ with A the cross section area. As stated before we transform to the fixed domain $\Omega_{ref} = [0,1] \times A$ using the transformation $\xi = \Psi(t,x) = x/L(t)$. This method is often referred to as Landau transformation [6] or boundary immobilization method [15]. This yields

$$\partial_t \bar{C} + \left(\frac{u}{L(t)} - \xi\frac{L'(t)}{L(t)}\right)\partial_\xi \bar{C} - \frac{D}{L(t)^2}\partial^2_{\xi\xi}\bar{C} = -\left(\partial_t \bar{M} - \xi\frac{L'(t)}{L(t)}\partial_\xi \bar{M}\right), \tag{28.22}$$

$$\partial_t \bar{M} - \xi\frac{L'(t)}{L(t)}\partial_\xi \bar{M} = \alpha|u|\bar{C}, \tag{28.23}$$

where $\xi \in [0,1]$. Note that the equation for $\bar{M}$ is now a PDE whereas we had an ODE for M. Therefore also a boundary condition on $\bar{M}$ needs to be imposed and, as described in [3], we set $\bar{M}(1,t) = 0$. This also reflects the physics that there is no deposition directly at the surface (M and accordingly $\bar{M}$ just measures deposition caused by depth filtration).

To solve the above problem we first compute the thickness $L(t)$, since the computation in (28.5) is decoupled from $\bar{C}$. Thus all time-dependent prefactors are known and the system for $\bar{C}$ and $\bar{M}$ can be solved with this knowledge. As the prefactors are scalar, they can be factored out from the system matrices and the offline/online decomposition can be used to compute the reduced matrices.

For our study we consider the velocity u and the initial concentration C_{in} as parameters and define the parameter domain $\mathscr{D} = [u_{up}, u_{down}] \times [C_{in,down}, C_{in,up}] \subset \mathbb{R}^2$ depicted in Fig. 28.2. We have a spherical particle mixture with $N_p = 117$ different particle sizes, which means that Eqs. (28.22) and (28.23) are repeated 117 times (with differing parameters). The cake height equations (28.5) are summed for

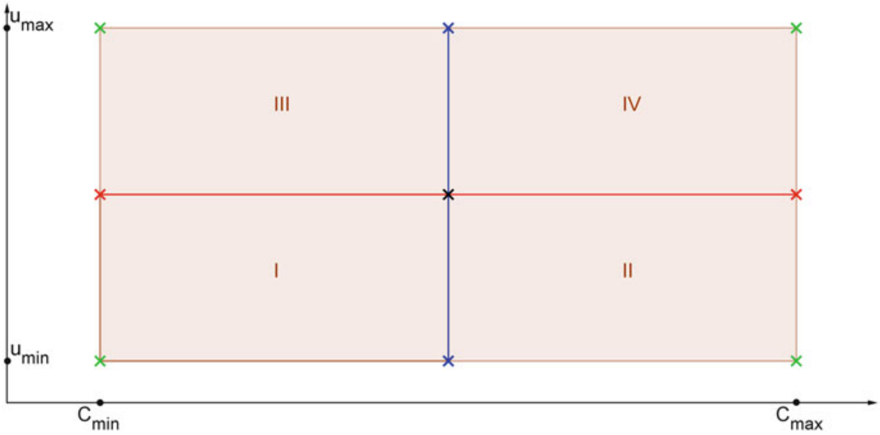


Fig. 28.2 Parameter domain $\mathscr{D}$ with interpolation points and subdomains

all particle sizes, i.e.,

$$\frac{\partial L}{\partial t} = \frac{|u|}{\rho_s(1-\phi_{cake})} \sum_{i=1}^{N_p} \lambda_i C_{in}^i, \tag{28.24}$$

where λ_i and C_{in}^i are the cake growth factor and the initial concentration for particle size i, respectively. For comparison we consider six different methods:

- *old/fixed/baseline* [7, 12]: The basis from one reference parameter configuration is reused for all other parameter configurations without adjustment.
- *optimal* [27]: For every parameter configuration the full order system is solved and the (optimal) POD basis is computed.
- *extrapolation* [1, 7, 8, 12]: From one reference point the POD modes and the corresponding sensitivities are computed. Then the current POD basis for a new parameter configuration is approximated by (28.19).
- *expansion* [1, 7, 12]: POD basis and the corresponding sensitivities are computed from one reference point. The sensitivities are added to the basis. An additional re-orthogonalization is necessary. For our example the size of the basis is 3ℓ.
- *interpolation* [2, 24]: For multiple points the full order system is solved and the POD basis is computed. Then the interpolation procedure from [2] is used with bilinear interpolation.
- *global* [22]: The full order solutions for multiple parameter configurations are merged into one snapshot matrix and a POD basis from all the information is computed. We choose the parameter values a priori and no Greedy algorithm for computing the optimal snapshot location is considered (for Greedy see e.g. [1]).

Below we refer to the methods whenever the name is written in italic script. Methods using the sensitivities directly in the snapshot matrix were tested for the above problem in [17]. The methods *old* and *optimal* serve as upper and lower bounds for the relative error in the tests. *Optimal* can be seen as the best achievable approximation with a given parameter configuration and a preset number of POD modes (lower error bound), whereas *old* can be seen as the easiest achievable approximation for any parameter configuration using a reference configuration (upper error bound). Thus an improved method should lie in between these bounds.

28.5 Numerical Tests

A cell-centered finite volume scheme with upwind discretization of the convective part and central differences for the diffusive part in space and implicit Euler scheme in time is used. The spatial resolution is $N = 8$ cells and the time step $\Delta t = 1\,\mathrm{s}$. From the solution of the discretized linear system the POD basis is computed.

The parameter domain is shown in Fig. 28.2, where the middle point (black cross) shows the reference point. The methods *extrapolation* and *expansion* use information (solution and sensitivities) from this point, i.e., $N_s = 1$. For *interpolation* the domain is divided in four subdomains. Here the four corner points are used as interpolation nodes for bilinear interpolation. Therefore the POD basis has to be known in all of the points. This method can be interpreted as a local method, since different bases are used in the subdomains (compare the classification in [18]). For the global basis, information from all nine depicted points ($N_s = 9$) are used.

The results are shown at two cuts through the parameter domain. The blue line reflects constant inflow with varying velocity and the red line vice versa. For both parameters 101 sampling points are used (total of 101^2 different parameter configurations). All other parameter values are listed in Table 28.1.

In the following we compare the relative error of the approximation (in comparison to the full order numerical solution) and the relative computation times (in comparison to the time for a full order numerical solve). We consider three different test cases: short time horizon, long time horizon and short time training.

For a short time horizon of $T_{end} = 60\,\mathrm{s}$ we have a total amount of 112320 unknowns. For the POD approximation we used three modes (i.e., 180 unknowns, approx. 0.1% of original amount of unknowns). Looking at the results for the cuts through the parameter domain in Fig. 28.3 one can see the following: The methods based on the reference data from the midpoint are coinciding in this point with the *optimal* approximation, since they are exactly designed in this

Table 28.1 Parameter values

$[u_{up}, u_{down}]$	$-0.002503235\,\mathrm{m/s} \times [1.5, 0.5]$	L_0	0.000691 m
$[C_{in,down}, C_{in,up}]$	$0.0048\,\mathrm{kg/m^3} \times [0.5, 1.5]$	D	$10^{-6}\mathrm{m^2/s}$
α	0–$25 \times 10^3\,\mathrm{m^{-1}}$	ρ_s	$2650\,\mathrm{kg/m^3}$
λ	0–1	ϕ_{cake}	0.5

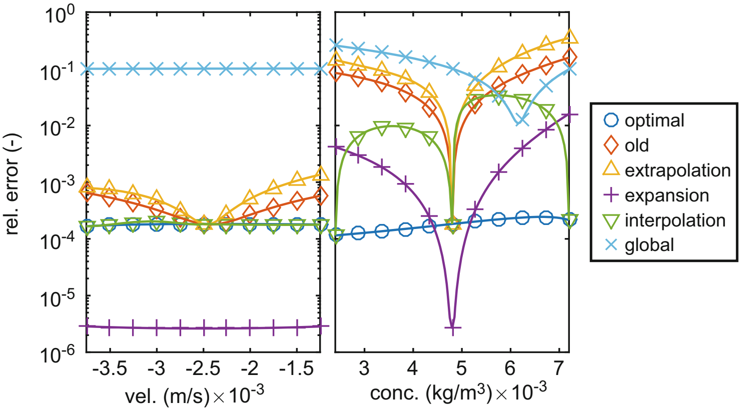


Fig. 28.3 Relative error for $T_{end} = 60\,\mathrm{s}$ and $\ell = 3$ POD modes, *right* variation of velocity, *left* variation of initial concentration

Table 28.2 Relative computation times for $T_{end} = 60\,\mathrm{s}$ and $\ell = 3$

	Old	Extrapolation	Expansion	Interpolation	Global
Offline cost	1.93	27.55	27.56	10.49	7.57
Av. online cost	0.31	0.31	0.32	0.31	0.31

way. For *interpolation* this is additionally true for the boundary points, since they are interpolation nodes. Comparing the approximation for velocity and inflow concentration one can see that the inflow concentration has a larger influence on the solution, since the variation in the relative error is much higher than for velocity. As mentioned before the number of unknowns in the reduced system is three times for *expansion* (i.e. 9) and therefore the approximation is better, especially in the reference point. For the velocity this is true in the whole parameter range, whereas for the inflow concentration this is just valid in a neighborhood of the reference point.

Looking at the previously defined bounds *old* (diamond) and *optimal* (circle), just *interpolation* and *expansion* lie in between and therefore yield an improvement of the approximation. The *global* approximation yields an overall balanced error for the whole parameter range, since it is built from several points and therefore has to approximate all points.

Comparing the relative computation times (Table 28.2) shows that the offline time can be very large (up to 28 times for *extrapolation* and *expansion*), whereas the average online cost are reduced to one-third of the average solution cost (three reduced order solves instead of one full order solve). Even if the reduced system is in the *expansion* case three times as large, the computational time is comparable due to the overall small size of the reduced system.

Let us shortly have a look at the relative error over the whole parameter domain shown in Fig. 28.4. Here one can see that *interpolation* and *optimal* coincide in all interpolation nodes. For all methods, which are just considering the information from the midpoint, the error is increasing towards the boundary of the parameter domain. As pointed out above, the concentration has a larger influence on the error. Taking a fixed concentration value and varying the velocity, the error is nearly constant, see Fig. 28.5. This is valid over nearly the whole domain. Fixing the

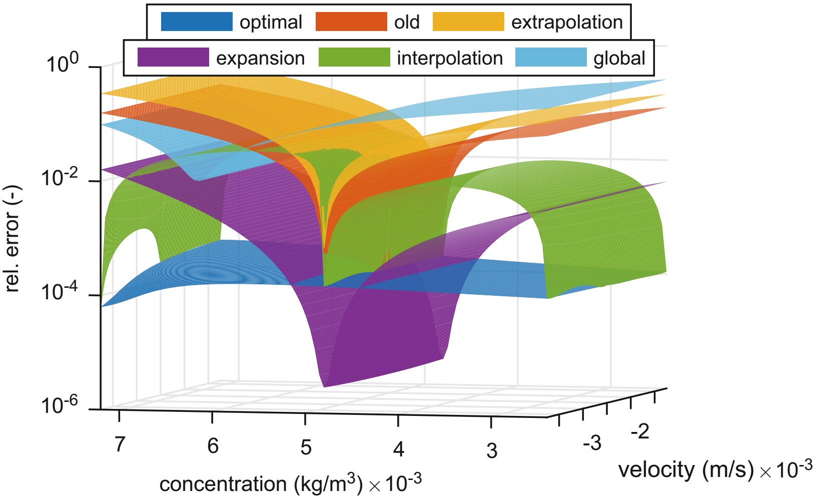


Fig. 28.4 Relative error for $T_{end} = 60\,\text{s}$ and $\ell = 3$ POD modes over the whole parameter domain

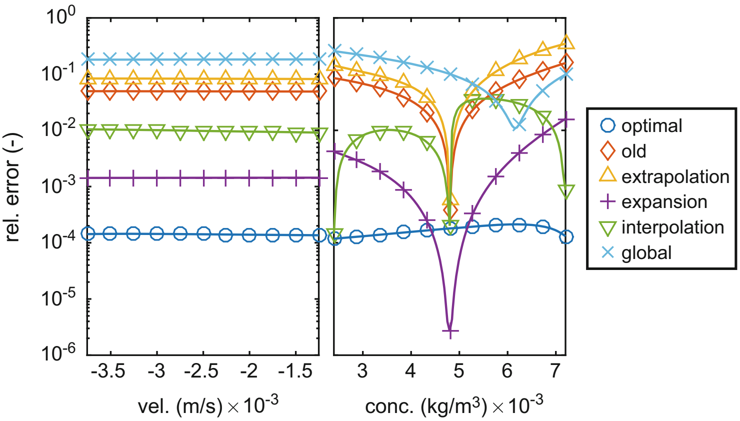


Fig. 28.5 Relative error for $T_{end} = 60\,\text{s}$ and $\ell = 3$ POD at off reference points, on the *left* $C_{in} = 0.0036\,\text{kg/m}^3$ with varying velocity and on the *right* $u = -0.00188\,\text{m/s}$ with varying concentration

velocity and varying the concentration, the behavior in Figs. 28.3 and 28.5 is similar, almost the same. For the cuts in Fig. 28.5 again *interpolation* and *expansion* lie in between the bounds, but, as can be seen in Fig. 28.4, one can also identify regions, where the *global* approximation lies in this range.

In the case of a long time horizon of $T_{end} = 3600$ s the total amount of unknowns in the full order system is 6739200 and 5 POD modes are used to approximate the solution (leading to 1800 unknowns in the reduced system). Considering the relative error in Fig. 28.6 one can see that the results are similar to the short time horizon. Again *interpolation* and *expansion* (with larger basis) yield the best approximation results.

Comparing the computation times (Table 28.3) one can see that the offline times decrease a lot, in particular for *extrapolation* and *expansion*. This indicates that the relative time to compute the sensitivities of the POD modes for a larger problem is in comparison not as large as for the example before. For all methods the average online computation times are below 10%.

For the last example we considered short time training for the long time horizon, i.e., we computed the solution for $T_{end} = 60$ s and used the derived POD modes for building the reduced system for $T_{end} = 3600$ s. The results are shown in Fig. 28.7 for $\ell = 6$ POD modes. The *global* approximation is left out, because the results were not satisfactory leading to a large approximation error. But another approximation is considered namely the *short time optimal* approximation. For every new parameter

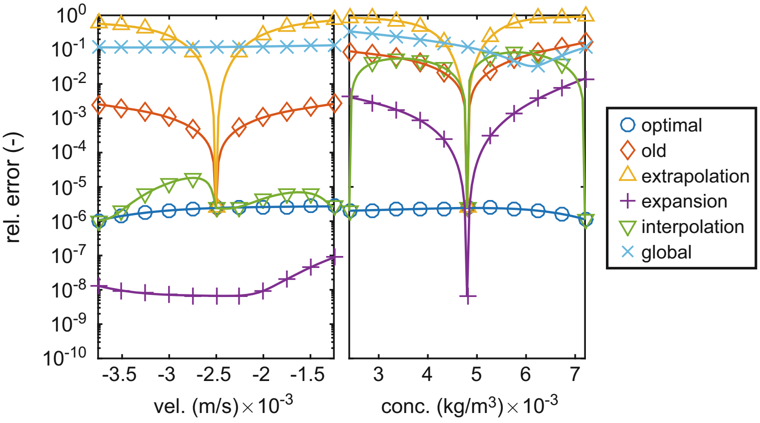


Fig. 28.6 Relative error for $T_{end} = 3600$ s and $\ell = 5$ POD modes, *right* variation of velocity, *left* variation of initial concentrations

Table 28.3 Relative computation times for $T_{end} = 3600$ s and $\ell = 5$

	Old	Extrapolation	Expansion	Interpolation	Global
Offline cost	1.00	3.95	3.95	8.22	8.19
Av. online cost	0.07	0.07	0.07	0.08	0.07

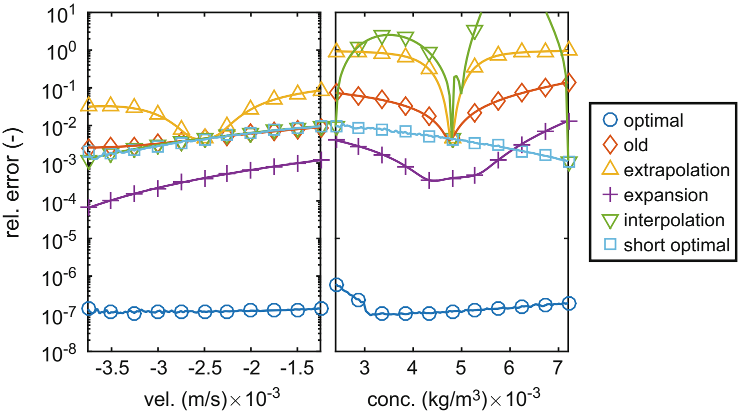


Fig. 28.7 Relative error for short time training and $\ell = 6$

Table 28.4 Relative computation times for short time training and $\ell = 6$

	Old	Extrapolation	Expansion	Interpolation	Short optimal
Offline cost	0.07	1.01	1.01	0.38	0.00
Av. online cost	0.06	0.07	0.09	0.07	0.11

configuration the full order solution for the short time horizon is computed and the optimal POD based is determined. This basis is used for the long time horizon as an approximation. Note that this method has no offline phase, since all computations are depending on the actual parameter values.

Looking at the results one can see that although the approximation is based on the short time solution, the results are good. The relative error is quite small and especially *expansion* yields good results, which are better than the *short time optimal* approximation. For the variation of velocity *interpolation* performs quite well, whereas in the case of initial concentration the relative error is larger than 1. Note that also the *old* approximation is computed from the short time horizon. For this example the computation times are most interesting (Table 28.4). One can see that also the relative offline cost are now decreased lot (here comparison is carried out with full order solve for long time horizon), but the average online cost are still on the same level as for the long time horizon case. Therefore here the speed up is achieved in the offline phase.

28.6 Conclusions

In this work we considered a model for combined depth and cake filtration. This model is subject to parameter variation and therefore a study for two parameters, namely the velocity and the inflow concentration, was carried out in 1D. For this purpose, different model reduction techniques based on POD, sensitivity analysis, and interpolation are used to speed up the computations. The numerical results show that this methods can speed up the computations. Of course this is just valid for the online stage, whereas in the offline phase a larger amount of computations has to be done. From the tested methods in particular *interpolation* and *expansion* are suitable to achieve a computational speed up in combination with a low approximation error. For short time training also the *short optimal* approximation has shown good results. The approximation error for *extrapolation* is larger than our upper bound in all the cases and therefore this method should not be used for our application.

References

1. Akman, T.: Local improvements to reduced-order approximations of optimal control problems governed by diffusion-convection-reaction equation. Comput. Math. Appl. **70**(2), 104–131 (2015)
2. Amsallem, D., Farhat, C.: Interpolation method for adapting reduced-order models and application to aeroelasticity. AIAA J. **46**(7), 1803–1813 (2008)
3. Breward, C.J.W., Byrne, H.M., Lewis, C.E.: A multiphase model describing vascular tumour growth. Bull. Math. Biol. **65**(4), 609–640 (2003)
4. Carlberg, K., Farhat, C.: A compact proper orthogonal decomposition basis for optimization-oriented reduced-order models. In: The 12th AIAA/ISSMO Multidisciplinary Analysis Optimization Conference, vol. 5964 (2008)
5. Carlberg, K., Farhat, C.: A low-cost, goal-oriented 'compact proper orthogonal decomposition' basis for model reduction of static systems. Int. J. Numer. Methods Eng. **86**(3), 381–402 (2011)
6. Crank, J.: Free and Moving Boundary Problems. Clarendon, Oxford (1984)
7. Hay, A., Borggaard, J.T., Pelletier, D.: Local improvements to reduced-order models using sensitivity analysis of the proper orthogonal decomposition. J. Fluid Mech. **629**, 41–72 (2009)
8. Hay, A., Akhtar, I., Borggaard, J.T.: On the use of sensitivity analysis in model reduction to predict flows for varying inflow conditions. Int. J. Numer. Methods Fluids **68**(1), 122–134 (2012)
9. Herzig, J.P., Leclerc, D.M., Goff, P.Le.: Flow of suspensions through porous media – application to deep filtration. Ind. Eng. Chem. **62**(5), 8–35 (1970)
10. Iliev, O., Kirsch, R., Osterroth, S.: Cake filtration simulation for poly-dispersed spherical particles. In: Proceedings Filtech 2015 Conference. L10-03-P112 (2015)
11. Iliev, O., Kirsch, R., Lakdawala, Z., Rief, S., Steiner, K.: Modeling and simulation of filtration processes. In: Currents in Industrial Mathematics: From Concepts to Research to Education, pp. 163–228. Springer, Berlin, Heidelberg (2015)
12. Jarvis, C.: Sensitivity based proper orthogonal decomposition for nonlinear parameter dependent systems. In: 2014 American Control Conference, pp. 135–140 (2014)
13. Kuhn, M., Briesen, H.: Dynamic modeling of filter-aid filtration including surface- and depth-filtration effects. Chem. Eng. Technol. **39**(3), 425–434 (2016)
14. Kunisch, K., Volkwein, S.: Galerkin proper orthogonal decomposition methods for parabolic problems. Numer. Math. **90**(1), 117–148 (2001)
15. Kutluay, S., Bahadir, A.R., Özdeş, A.: The numerical solution of one-phase classical Stefan problem. J. Comput. Appl. Math. **81**(1), 135–144 (1997)
16. Ni, L.A., Yu, A.B., Lu, G.Q., Howes, T.: Simulation of the cake formation and growth in cake filtration. Miner. Eng. **19**(10), 1084–1097 (2006)
17. Osterroth, S., Iliev, O., Pinnau, R.: A combined sensitivity analysis and model reduction workflow for the simulation of cake filtration. In: Young Researchers Symposium, YRS 2016. Proceedings, pp. 115–120 (2016)

18. Peng, L., Mohseni, K.: Nonlinear model reduction via a locally weighted POD method. Int. J. Numer. Methods Eng. **106**(5), 372–396 (2016)
19. Pinnau, R.: Model reduction via proper orthogonal decomposition, In: Model Order Reduction: Theory, Research Aspects and Applications, pp. 95–109. Springer, Berlin, Heidelberg (2008)
20. Prud'homme, C., Rovas, D.V., Veroy, K., Machiels, L., Maday, Y., Patera, A.T., Turinici, G.: Reliable real-time solution of parametrized partial differential equations: reduced-basis output bound methods. J. Fluids Eng. **124**(1), 70–80 (2002)
21. Quarteroni, A., Manzoni, A., Negri, F.: Reduced Basis Methods for Partial Differential Equations: An Introduction, vol. 92. Springer, Berlin (2015)
22. Schmit, R., Glauser, M.: Improvements in low dimensional tools for flow-structure interaction problems: Using global POD. In: Proceedings of the 42nd AIAA Aerospace Science Meeting and Exhibit (2004)
23. Schmidt, A., Potschka, A., Körkel, S., Bock, H.G.: Derivative-extended POD reduced-order modeling for parameter estimation. SIAM J. Sci. Comput. **35**(6), A2696–A2717 (2013)
24. Son, N.T.: A real time procedure for affinely dependent parametric model order reduction using interpolation on Grassmann manifolds. Int. J. Numer. Methods Eng. **93**(8), 818–833 (2013)
25. Stamatakis, K., Tien, C.: Cake formation and growth in cake filtration. Chem. Eng. Sci. **46**(8), 1917–1933 (1991)
26. Tien, C.: Introduction to Cake Filtration: Analyses, Experiments and Applications. Elsevier, Amsterdam (2006)
27. Volkwein, S.: Model reduction using proper orthogonal decomposition. Lecture Notes, University of Graz (2011). http://www.math.uni-konstanz.de/numerik/personen/volkwein/teaching/POD-Vorlesung.pdf. Accessed on 13 May 2016

Chapter 29
Model Reduction for Coupled Near-Well and Reservoir Models Using Multiple Space-Time Discretizations

Walid Kheriji, Yalchin Efendiev, Victor Manuel Calo, and Eduardo Gildin

Abstract In reservoir simulations, fine fully-resolved grids deliver accurate model representations, but lead to large systems of nonlinear equations to solve every time step. Numerous techniques are applied in porous media flow simulations to reduce the computational effort associated with solving the underlying coupled nonlinear partial differential equations. Many models treat the reservoir as a whole. In other cases, the near-well accuracy is important as it controls the production rate. Near-well modeling requires finer space and time resolution compared with the remaining of the reservoir domain. To address these needs, we combine Model Order Reduction (MOR) with local grid refinement and local time stepping for reservoir simulations in highly heterogeneous porous media. We present a domain decomposition algorithm for a gas flow model in porous media coupling near-well regions, which are locally well-resolved in space and time with a coarser reservoir discretization. We use a full resolution for the near-well regions and apply MOR in the remainder of the domain. We illustrate our findings with numerical results on a gas flow model through porous media in a heterogeneous reservoir.

W. Kheriji (✉)
Texas A&M University at Qatar, PO Box 23874, Education City Doha, Qatar
e-mail: kheriji.walid@gmail.com

Y. Efendiev • E. Gildin
Texas A&M University, College Station, TX 77843, USA
e-mail: efendiev@math.tamu.edu; egildin@tamu.edu

V.M. Calo
Mineral Resources, CSIRO, Curtin University, Kensington, WA, Australia
e-mail: victor.calo@curtin.edu.au

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_29

29.1 Introduction

Proper reservoir management often is challenging to perform due to the intrinsic uncertainties and complexities associated with the reservoir properties (see [31]). To this end, accurate results for reservoirs are obtained if a fully-resolved, fine grid discretization is used in the model. At every time step, this requires the solution of large systems of nonlinear equations. The importance of obtaining a simpler model that can represent the physics of the full system is paramount to speed up the workflows that require many (from dozens to thousands) calls of the forward model. This is usually the case in history matching (see [1, 27]), production optimization problems (see [10]) and uncertainty quantifications (see [23]). Also, the computational time of such large-scale models become the bottleneck of fast turnarounds in the decision-making process and assimilation of real-time data into reservoir models (see [16, 21]). Over the past decade, numerous techniques have been applied in porous media flow simulation to reduce the computational effort associated with the solution of the underlying coupled nonlinear partial differential equations. These techniques range from heuristic approaches (see [25, 29]), to more elegant mathematical techniques (see [3, 22]), explore the idea of reducing the complexity of a model that can approximate the full nonlinear system of equations with controlled accuracy. In many cases, reduced-order modeling techniques are a viable way of mitigating computational cost when simulating a large scale model, while they maintain high accuracy when compared with high fidelity models. Reduced order modeling by projection has been used in systems/controls, framework, such as the balanced truncation (see [22]), proper orthogonal decomposition (POD) (see [11]), the trajectory piecewise linear (TPWL) techniques of Cardoso and Durlofsky [6], empirical interpolation methods (see [12, 19]), bilinear Krylov subspace methods (see [17]) and quadratic bilinear model order reduction (see [20]). Many of these simulation models treat the reservoir as a whole model, while near-well regions in reservoir simulations usually require Local Grid Refinement (LGR) and Local Time Stepping (LTS) due to several physical processes that occur in these regions such as higher Darcy velocities, the coupling of the stationary well model with the transient reservoir model, high non-linearities due to phase segregation (typically gas separates) and complex physics such as formation damage models. In addition the near-well geological model is usually finer in the near-well region due to the higher availability of reliable data. Different approaches combining LTS and LGR have been studied for reservoir simulation applications. The first class of algorithms belongs to Domain Decomposition Methods (DDM). Matching conditions are defined at the near-well reservoir interface with possible overlap, and a Schwarz algorithm is used to compute the solution (see [13, 26]). A second class of methods uses both a coarse grid on the full domain and a LGR in the near-well grid (usually called windowing). These grids communicate both at the near-well reservoir interface and also between the perforated fine and coarse cells. In [24], Walid et al. combined these two latter approaches. An efficient iterative algorithm is obtained using at the near-well reservoir interface, optimized Robin conditions for the pressure. DDM and MOR has been combined and applied in different multiphysics problems (see [2, 4, 5, 8, 9], and [30]).

In this chapter, we combine the DDM algorithm developed in [24] with the MOR technique developed in [18]. We describe model reduction techniques that consider near-well and reservoir regions separately and use different spatial and temporal resolutions to achieve efficient and accurate reduced order models. We use full resolution to solve the near-well discretization and apply MOR (POD-DEIM) in the rest of the domain. We use POD to construct a low-order model using snapshots formed from a forward simulation with the original high-order model. In the presence of a general nonlinearity, the computational complexity of the reduced model still depends to the original fully-resolved discretization. By employing the Discrete Empirical Interpolation Method (DEIM), we reduce the computational complexity of the nonlinear term of the reduced model to a cost proportional to the number of reduced variables obtained by POD.

This chapter is organized as follows. We first present in Sect. 29.2 a compressible flow model in porous media. Then, in Sect. 29.3 we describe the local space and time refinement discretization coupled with model order reduction using POD and DEIM. Finally in Sect. 29.4 we illustrate the efficiency of our MOR-DDM algorithm on 2D test cases both in terms of accuracy and CPU time compared with the reference solution obtained using the LGR grid with global fine time stepping and full resolution.

29.2 Compressible Flow Model in Porous Media

In this section we consider compressible phase flow in a porous media. The model describes the injection of gas through a injector well in a 2D reservoir initially saturated with gas. The velocity is given by the Darcy laws

$$\mathbf{V} = -\frac{1}{\mu}\,\mathbf{K}\,\nabla p, \tag{29.1}$$

where p is the pressure and μ is the gas viscosity assumed to be constant. The rock permeability is denoted by $\mathbf{K}$ and the rock porosity by ϕ. Then, the pressure p is solution of the following mass conservation equation.

$$\begin{cases} \phi\ \partial_t \rho(p) + \ \boldsymbol{\nabla}\cdot(\rho(p)\,\mathbf{V}) = 0, & \text{in } \Omega_r \times (0,T), \\ -\mathbf{K}\nabla p\cdot\mathbf{n} = 0, & \text{on } \Gamma_r \times (0,T), \\ p = p_{\text{bhp}}, & \text{on } \Gamma_w \times (0,T), \\ p = p_{\text{init}}, & \text{in } \Omega_r \times \{0\}, \end{cases} \tag{29.2}$$

where $\rho(p)$ is the mass density (assumed linear) and $p_{\text{bhp}}(t)$ is the imposed bottom hole pressure at the well boundary. To simplify notation, we assume that the

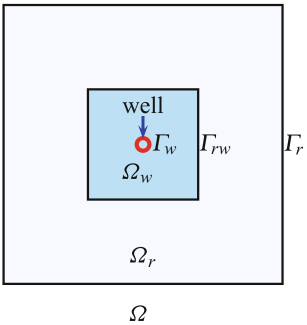


Fig. 29.1 Example of reservoir domain Ω_r and near-well subdomain Ω_w with the near-well reservoir interface Γ_{rw}, and the well boundaries Γ_w

injection pressure $p_{\mathrm{bhp}}(t)$ is chosen such that $-\mathbf{K}\nabla p \cdot \mathbf{n}_w < 0$ at the well boundaries Γ_w, where $\mathbf{n}_w$ is the unit normal vector at the well boundaries outward to Ω_r. The case of producer wells could also be dealt without additional difficulties. The near-well accuracy controls the injection (production) rate which motivates the use of a near-well refinement of the spatial and temporal resolution for the simulation of this model.

Let us denote by $\Omega_w \subset \Omega_r$ the near-well region. In the following, the outer boundary of the near-well region Ω_w is denoted by Γ_{rw} (see Fig. 29.1). We use a model order reduction-domain decomposition method (MOR-DDM) to solve Eqs. (29.1)–(29.2) with a coarse discretization in space and time in the reservoir domain Ω_r and a locally refined space and time discretization in the near-well region Ω_w. These discretizations are coupled by solving iteratively both subproblems on a given time interval (t^{n-1}, t^n) using appropriate interface conditions at Γ_w and Γ_{rw}. A Robin condition for the pressure is used at the boundary Γ_{rw} of the subdomain Ω_w. At the well boundary Γ_w of the domain Ω_r, a total flux Neumann condition is imposed.

29.3 Model Order Reduction Using Local Space and Time Refinement

Instead of using a local grid refinement and a global fine time step size with full resolution to solve Eqs. (29.1)–(29.2), we use a domain decomposition method coupling the coarse discretization in space and time in the reservoir domain using POD-DEIM with a fine discretization in space and time in the near-well domain using full resolution. In the following, first the coarse and fine finite volume discretizations of Ω_r and Ω_w are introduced, then we describe the MOR-DDM algorithm with a single time step, and finally the extension taking into account local time stepping schemes in the near-well domain is explained.

29.3.1 Two Level Finite Volume Discretization

The discretization (see Fig. 29.2) starts from a coarse finite volume mesh of the full reservoir domain Ω_r defined by

$$\left(\mathscr{M}_r, \mathscr{F}_r^{\text{int}}, \mathscr{P}_w\right),$$

where $\mathscr{M}_r$ is the set of coarse cells K, $\mathscr{F}_r^{\text{int}}$ the set of coarse inner faces σ, and $\mathscr{P}_w$ the set of well perforations. The mesh is assumed to be conforming in the sense that the set of neighbouring cells $\mathscr{M}_\sigma \subset \mathscr{M}_r$ of an inner face $\sigma \in \mathscr{F}_r^{\text{int}}$ contains exactly two cells K and L. The inner face σ is denoted by $\sigma = K|L$. Considering that the size of the cells is very large compared with the well radius, the wells are discretized using Peaceman's indices in each perforated cell [28]. For the sake of simplicity, the well is assumed to be vertical with consequently, in our horizontal 2D case, a single perforation. Let us denote by $\mathscr{P}_w$ the set of perforations σ and by $K_\sigma^r \in \mathscr{M}_r$ the corresponding perforated coarse cells.

A set of near-well coarse cells is assumed to be refined (coarse cells inside the red boundary in Fig. 29.2) and the near-well mesh is obtained by adding a layer of

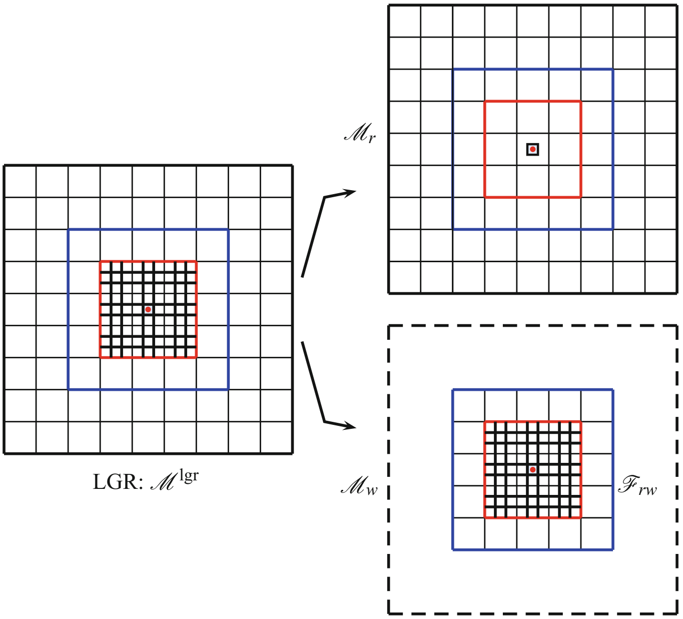


Fig. 29.2 *Left*: LGR mesh $\mathscr{M}^{\text{lgr}}$. *Top right*: reservoir coarse mesh $\mathscr{M}_r$. *Bottom right*: near-well fine mesh $\mathscr{M}_w$ and near-well reservoir interfaces $\mathscr{F}_{rw}$

coarse cells at the boundary of the union of the refined cells. The resulting near-well mesh is defined by

$$\left(\mathscr{M}_w, \mathscr{F}_w^{\text{int}}, \mathscr{F}_{rw}, \mathscr{P}_w\right),$$

where $\mathscr{M}_w$ is the set of cells K, $\mathscr{F}_w^{\text{int}}$ is the set of inner faces σ, $\mathscr{P}_w$ is the set of well perforations, and $\mathscr{F}_{rw} \subset \mathscr{F}_r^{int}$ is the set of boundary faces corresponding by construction to coarse faces. The fine perforated cells are denoted by K_σ^w for all perforations $\sigma \in \mathscr{P}_w$ as Fig. 29.2 displays. We assume that the near-well mesh is conforming in the sense that the set of neighboring cells $\mathscr{M}_\sigma \subset \mathscr{M}_w$ of an inner face $\sigma \in \mathscr{F}_w^{\text{int}}$ contains exactly two cells K and L, and the inner face σ is denoted by $\sigma = K|L$. At the near-well reservoir interface, for each face $\sigma \in \mathscr{F}_{rw}$, we assume that the set of the two neighboring cells $\mathscr{M}_\sigma = \{K, L\}$ is ordered such that $K \in \mathscr{M}_w \cap \mathscr{M}_r$ and $L \in \mathscr{M}_r \setminus \mathscr{M}_w$.

A cell centre finite volume discretization is used for the discretization of the compressible flow model. We will denote by $\mathbf{P}_r$ (resp. $\mathbf{P}_w$) the vector of cell pressures $\mathbf{P}_{r,K}$, $K \in \mathscr{M}_r$ (resp. $\mathbf{P}_{w,K}$, $K \in \mathscr{M}_w$). Let $\sigma = K|L$ be an inner coarse or fine face, and $\mathbf{n}_{K,\sigma}$ the unit normal vector at the face σ outward to the cell K. Let $\mathbf{P}$ be the reservoir or near-well discrete pressure $\mathbf{P}_r$ or $\mathbf{P}_w$. Assuming the orthogonality of the mesh w.r.t. the permeability field $\mathbf{K}$, the Darcy flux $\int_\sigma -\mathbf{K}\nabla p \cdot \mathbf{n}_{K,\sigma} d\sigma$ is approximated by the following conservative Two Point Flux Approximation (TPFA) [14]

$$F_{K,\sigma}(P) = T_\sigma(\mathbf{P}_K - \mathbf{P}_L),$$

where T_σ is the transmissivity of the face $\sigma \in \mathscr{F}_r^{int}$ or $\sigma \in \mathscr{F}_w^{int}$. A Two Point flux approximation of the Darcy flux is also assumed at the near-well reservoir interface $\sigma = K|L \in \mathscr{F}_{rw}$. It is denoted by

$$F_{K,\sigma}(\mathbf{P}_{w,K}, \mathbf{P}_{r,L}) = T_\sigma(\mathbf{P}_{w,K} - \mathbf{P}_{r,L}),$$

where T_σ is the transmissivity of the face σ. In the following DDM algorithm, $\mathbf{P}_{r,L}$ represents the pressure interface value viewed by the near-well subdomain in order to obtain the same finite volume discretization than the one obtained on the single LGR mesh $\mathscr{M}^{\text{lgr}}$ shown in Fig. 29.2. For each $\sigma \in \mathscr{P}_w$, the Darcy flux $\int_\sigma -\mathbf{K}\nabla p \cdot \mathbf{n}_{K,\sigma} d\sigma$ at the well perforation boundary is defined by the two point flux approximation

$$F_{K,\sigma}^s(\mathbf{P}_{s,K}, \mathbf{P}_\sigma) = PI_\sigma^s(\mathbf{P}_{s,K} - \mathbf{P}_\sigma),$$

where $\mathbf{P}_\sigma$ denotes the pressure inside the perforation, $K = K_\sigma^s$ is the coarse ($s = r$) or fine ($s = w$) perforated cell, and PI_σ^s for $s = r$ or $s = w$ is the modified transmissivity of the perforation σ in the cell K obtained using the Peaceman formula which takes into account the singularity of the pressure solution at the well (see [28]).

29.3.2 Model Order Reduction: Domain Decomposition Method

At the near-well reservoir interface $\mathscr{F}_{rw}$, a Robin optimized interface condition is used.

$$\lambda\, p_w + \alpha\, \mathbf{K}\, \nabla p_w \cdot \mathbf{n}_w = \lambda\, p_r + \alpha\, \mathbf{K}\, \nabla p_r \cdot \mathbf{n}_w, \tag{29.3}$$

where $\mathbf{n}_w$ is the normal at Γ_{rw} outward Ω_w, and λ and α are two positive optimized parameters (see [24]). The parameter α is set to 1 and the parameter λ is chosen to optimize the convergence rate leading to an optimized DD algorithm. The optimization of the coefficient λ is done using existing theory for optimized Schwarz methods (see [15]), the optimal parameter can be computed analytically in such a way that the DDM algorithm converges in two iterations after time integration on one coarse time step, without taking into account the local time stepping. On the well boundary Γ_w, a Neumann total flux condition is used. In our injection well example, we obtain the following condition

$$-\rho(p_r)\frac{1}{\mu}\mathbf{K}\nabla p_r = -\rho(p_{\mathrm{bhp}})\frac{1}{\mu}\mathbf{K}\nabla p_w, \tag{29.4}$$

Let us consider, on the reservoir and near-well meshes, the same time discretization $t^0, t^1, \cdots, t^N$ of the interval $(0, T)$ with $t^0 = 0, t^N = T$, and $\Delta t^n = t^n - t^{n-1} > 0$, $n = 1, \cdots, N$. The gas flow model in porous media is integrated by an implicit Euler scheme. The discretization in space uses the TPFA discretization of the Darcy flow together with an upwinding of the mass density with respect to the sign of the Darcy flow. Let the reservoir and near-well solutions at time t^{n-1} be given. Let us denote by $x^+ = \max(x, 0)$ and $x^- = \min(x, 0)$. Then, knowing the near-well solution $\mathbf{P}_w$ at time t^n, the reservoir subproblem computes the solution $\mathbf{P}_r$ at time t^n of the conservation equations is given by:

$$\begin{cases} \text{For each cell } K \in \mathscr{M}_r, \\[1ex] \phi_K \dfrac{|K|}{\Delta t^n}\Big(\rho(\mathbf{P}_{r,K}) - \rho(\mathbf{P}^{n-1}_{r,K})\Big) \\[1ex] + \displaystyle\sum_{\sigma=K|L\in\mathscr{F}_r^{\mathrm{int}}} \frac{\rho(\mathbf{P}_{r,K})}{\mu} F_{K,\sigma}(\mathbf{P}_r)^+ + \sum_{\sigma=K|L\in\mathscr{F}_r^{\mathrm{int}}} \frac{\rho(\mathbf{P}_{r,L})}{\mu} F_{K,\sigma}(\mathbf{P}_r)^- \\[1ex] + \displaystyle\sum_{\sigma\in\mathscr{P}_w\,|\,K^r_\sigma=K} \frac{\rho(p_{\mathrm{bhp}})}{\mu} F^r_{K,\sigma}(\mathbf{P}_{r,K}, p_{\mathrm{bhp}}) = 0, \end{cases} \tag{29.5}$$

coupled with the well perforations interface conditions for all $\sigma \in \mathscr{P}_w$

$$\frac{\rho(p_{\text{bhp}})}{\mu} F^r_{K,\sigma}(\mathbf{P}_{r,K}, p_{\text{bhp}}) = \frac{\rho(p_{\text{bhp}})}{\mu} F^w_{K,\sigma}(\mathbf{P}_{w,K^w_\sigma}, p_{\text{bhp}}). \tag{29.6}$$

Knowing the solution $\mathbf{P}_r$, the near-well subproblem computes the solution $\mathbf{P}_w$, of the conservation equations is given by:

$$\left\{\begin{aligned} &\text{For each cell } K \in \mathscr{M}_w, \\ &\phi_K\Big(\rho(\mathbf{P}_{w,K}) - \rho(\mathbf{P}^{n-1}_{w,K})\Big)\frac{|K|}{\Delta t^n} \\ &+ \sum_{\sigma=K|L \in \mathscr{F}^{\text{int}}_w} \frac{\rho(\mathbf{P}_{w,K})}{\mu} F_{K,\sigma}(\mathbf{P}_w)^+ + \sum_{\sigma=K|L \in \mathscr{F}^{\text{int}}_w} \frac{\rho(\mathbf{P}_{w,L})}{\mu} F_{K,\sigma}(\mathbf{P}_w)^- \\ &+ \sum_{\sigma=K|L \in \mathscr{F}_{rw}} \frac{\rho(\mathbf{P}_{w,K})}{\mu} F_{K,\sigma}(\mathbf{P}_{w,K}, \mathbf{P}_{w,\sigma})^+ + \sum_{\sigma=K|L \in \mathscr{F}_{rw}} \frac{\rho(\mathbf{P}_{w,\sigma})}{\mu} F_{K,\sigma}(\mathbf{P}_{w,K}, \mathbf{P}_{w,\sigma})^- \\ &+ \sum_{\sigma \in \mathscr{P}_w \,|\, K^w_\sigma = K} \frac{\rho(p_{\text{bhp}})}{\mu} F^w_{K,\sigma}(\mathbf{P}_{w,K}, p_{\text{bhp}}) = 0, \end{aligned}\right. \tag{29.7}$$

coupled with the following near-well reservoir interface conditions for all $\sigma = K|L$

$$|\sigma|\lambda_\sigma \mathbf{P}_{w,\sigma} - \alpha_\sigma F_{K,\sigma}(\mathbf{P}_{w,K}, \mathbf{P}_{w,\sigma}) = |\sigma|\lambda_\sigma \mathbf{P}_{r,\sigma} - \alpha_\sigma F_{K,\sigma}(\mathbf{P}_{r,K}, \mathbf{P}_{r,\sigma}), \tag{29.8}$$

where $|\sigma|$ the lengh of the face σ.

Model reduction is performed using POD and DEIM, to solve the reservoir subproblem (29.5)–(29.6) coupled with a fully-resolved of the near-well subproblem (29.7)–(29.8). POD constructs a low-order model using snapshots from a forward simulation with the original high-order model using fine time step and LGR mesh $\mathscr{M}^{lgr}$.

Let us denote by n_r the number of cells in the mesh $\mathscr{M}_r$ located into the subdomain $\Omega_r \setminus \Omega_w$, by n_w the number of cells in the mesh $\mathscr{M}_r$ located in the near-well domain Ω_w, and by $n_p \ll n_r$ the reduced pressure dimensional space. Given a set of snapshots $\mathbb{S}_{\mathbf{P}} = \Big[\mathbf{P}_r(t^1), \mathbf{P}_r(t^2), \ldots, \mathbf{P}_r(t^N)\Big] \in \mathbb{R}^{n_r \times N}$, we apply a singular value decomposition (SVD) on the matrix $\mathbb{S}_{\mathbf{P}}$. The POD basis matrix $\varphi \in \mathbb{R}^{n_r \times n_p}$, corresponds to the first n_p left singular vectors. To extend the POD basis matrix to the full reservoir domain and to keep the full well-resolution in the near-well region, we define the following prolongation of the POD basis matrix

$$\tilde{\varphi} = \begin{pmatrix} \varphi & 0 \\ 0 & I_{n_w} \end{pmatrix} \in \mathbb{R}^{n_{rw} \times n_{pw}},$$

where $n_{rw} = n_r + n_w$ and $n_{pw} = n_p + n_w$, then the pressure is projected into the reduced subspace as, $\mathbf{P}_r(t) = \tilde{\varphi}\mathbf{p}_r(t)$, where $\mathbf{p}_r(t) \in \mathbb{R}^{n_{pw}}$ is the reduced solution. POD is usually limited to problems with linear or bilinear terms. In the presence of a general nonlinearity, the computational complexity of the reduced model still depends to the finely resolved discretization. DEIM effectively overcomes this shortcoming of the method. DEIM constructs a subspace to approximate the nonlinear terms and selects points that specify an interpolation based projection of dimension $m_p \ll n_r$ to give a nearly optimal subspace approximation to the nonlinear term (see [7, 18]).

Let us denote by $\mathbf{N}_r\ (\mathbf{P}_r(t))$ the nonlinear term in the reservoir subproblem (29.5)–(29.6), then for each cell $K \in \mathscr{M}_r$, $(\mathbf{N}_r\ (\mathbf{P}_r))_K$ is given by:

$$\left\{\begin{aligned} (\mathbf{N}_r\ (\mathbf{P}_r))_K = & \sum_{\sigma=K|L\in\mathscr{F}_r^{\text{int}}} \frac{\rho(\mathbf{P}_{r,K})}{\mu} F_{K,\sigma}(\mathbf{P}_r)^+ + \sum_{\sigma=K|L\in\mathscr{F}_r^{\text{int}}} \frac{\rho(\mathbf{P}_{r,L})}{\mu} F_{K,\sigma}(\mathbf{P}_r)^- \\ & + \sum_{\sigma\in\mathscr{P}_w\,|\,K_\sigma^r=K} \frac{\rho(p_{\text{bhp}})}{\mu} F_{K,\sigma}^r(\mathbf{P}_{r,K}, p_{\text{bhp}}) \end{aligned}\right. \tag{29.9}$$

Let us define the diagonal matrix $L = (|K|\phi_K)_{K\in\mathscr{M}_r} \in \mathbb{R}^{n_{rw}\times n_{rw}}$ where ϕ_K and $|K|$ denote, respectively, the porosity and the surface of the cell K. Then the system (29.5) can be rewritten in the following algebraic form:

$$\frac{1}{\Delta t^n}\mathbf{L}\left(\rho(\mathbf{P}_r) - \rho(\mathbf{P}_r^{n-1})\right) + \mathbf{N}_r\ (\mathbf{P}_r) = 0. \tag{29.10}$$

We replace $\mathbf{P}_r$ and $\mathbf{P}_r^{n-1}$ respectively by $\tilde{\varphi}\mathbf{p}_r$ and $\tilde{\varphi}\mathbf{p}_r^{n-1}$ and we project the system (29.10) onto $\tilde{\varphi}$, then the reduced system of (29.10) is of the form:

$$\frac{1}{\Delta t^n}\tilde{\varphi}^T\,\mathbf{L}\,\tilde{\varphi}\left(\rho(\mathbf{p}_r) - \rho(\mathbf{p}_r^{n-1})\right) + \tilde{\varphi}^T\,\mathbf{N}_r\ (\tilde{\varphi}\mathbf{p}_r) = 0, \tag{29.11}$$

We approximate the nonlinear function $\mathbf{N}_r$ on a linear subspace spanned by basis vectors $\Psi = \left(\psi_1, \cdots, \psi_{m_p}\right) \in \mathbb{R}^{n_r\times m_p}$, obtained by applying POD to the snapshots of the nonlinear function $\mathbf{N}_r$: $\mathbb{S}_{\mathbf{N}_r} = \left[\mathbf{N}_r(\mathbf{P}_r(t^1)), \mathbf{N}_r(\mathbf{P}_r(t^2)), \cdots, \mathbf{N}_r(\mathbf{P}_r(t^N))\right] \in \mathbb{R}^{n_r\times N}$. Similarly to POD, to extend the DEIM to the full reservoir domain and to keep the full well-resolution in the near-well region, we define the following prolongation of the DEIM basis matrix

$$\tilde{\Psi} = \begin{pmatrix} \Psi & 0 \\ 0 & I_{n_w} \end{pmatrix} \in \mathbb{R}^{n_{\text{lgr}}\times m_{pw}},$$

where $m_{pw} = m_p + n_w$, then $\mathbf{N}_r \approx \sum_{i=1}^{m_{pw}} c_i\, \tilde{\psi}_i = \tilde{\Psi}\mathbf{c}$. Thus, DEIM selects only m_{pw} rows of $\tilde{\Psi}$ to compute the coefficients $\mathbf{c}$. This can be formalized using the selection matrix

$$\mathscr{P} = \left[e_{\wp_1}, \ldots, e_{\wp_{m_{pw}}}\right] \in \mathbb{R}^{n_{\mathrm{lgr}} \times m_{pw}}$$

where e_i is the ith column of the identity matrix. Assume $\mathscr{P}^T\, \tilde{\Psi}$ is nonsingular, the reduced system (29.11) becomes

$$\frac{1}{\Delta t^n} \underbrace{\tilde{\varphi}^T\, \mathbf{L}\, \tilde{\varphi}}_{n_{pw}\times n_{pw}} \Big(\rho(\mathbf{p}_r) - \rho(\mathbf{p}_r^{n-1})\Big) + \underbrace{\tilde{\varphi}^T\, \tilde{\Psi}(\mathscr{P}^T\, \tilde{\Psi})^{-1}}_{n_{pw}\times m_{pw}} \underbrace{\mathscr{P}^T\, \mathbf{N}_r\, (\tilde{\varphi}\mathbf{p}_r)}_{m_{pw}\times 1} = 0, \quad (29.12)$$

When the nonlinearity is component-wise, the selection matrix $\mathscr{P}^T$ can be brought inside the nonlinearity $\mathbf{N}_r$ and hence the computational complexity of $\mathscr{P}^T\, \mathbf{N}_r\, (\tilde{\varphi}\mathbf{p}_r)$ is independent of the fine grid dimension n_{lgr} (size of high fidelity model). This is obviously not applicable in our case . However, thanks to the TPFA discretization, the evolution of each nonlinear element depends only to the neighboring elements, and therefore it is possible to compute the nonlinear term $\mathscr{P}^T\, \mathbf{N}_r\, (\tilde{\varphi}\mathbf{p}_r)$ independently of the fine grid dimension n_{lgr} using a certain sparse matrix data structure. Let $\tilde{\varphi}_K$ the row of the basis matrix $\tilde{\varphi}$ corresponding to the cell $K \in \mathscr{M}_r$, then $\mathbf{P}_{r,K} = \tilde{\varphi}_K \mathbf{p}_r$, and hence the Two Point flux approximation of the Darcy flux can be rewritten in the following reduced order form

$$F_{K,\sigma}(\mathbf{P_r}) = T_\sigma(\mathbf{P}_{r,K} - \mathbf{P}_{r,L}) = T_\sigma(\tilde{\varphi}_K - \tilde{\varphi}_L)\mathbf{p}_r = \tilde{F}_{K,\sigma}(\,\mathbf{p_r})$$

Using the notations above, Eq. (29.9) can be rewritten in the following reduced order form

$$\left\{ \begin{aligned} (\tilde{\mathbf{N}}_r\, (\,\mathbf{p}_r))_K = &\sum_{\sigma=K|L\in\mathscr{F}_r^{\mathrm{int}}} \frac{\rho(\tilde{\varphi}_K\mathbf{p}_r)}{\mu} \tilde{F}_{K,\sigma}(\,\mathbf{p}_r)^+ + \sum_{\sigma=K|L\in\mathscr{F}_r^{\mathrm{int}}} \frac{\rho(\tilde{\varphi}_L\mathbf{p}_r)}{\mu} \tilde{F}_{K,\sigma}(\,\mathbf{p}_r)^- \\ &+ \sum_{\sigma\in\mathscr{P}_w\,|\,K_\sigma^r=K} \frac{\rho(p_{r,\sigma})}{\mu} F_{K,\sigma}^r(\tilde{\varphi}_K\mathbf{p}_r, p_{r,\sigma}), \end{aligned} \right. \quad (29.13)$$

with $p_{r,\sigma}$ denotes the pressure inside the perforation. Let us denote by K^i the ith cell in the reservoir mesh $\mathscr{M}_r$. Then, a new formulation of $\mathbf{N}_r(\tilde{\varphi}\mathbf{p}_r)$ is provided by

$$\mathbf{N}_r\, (\tilde{\varphi}\mathbf{p}_r) = \left[(\tilde{\mathbf{N}}_r\, (\,\mathbf{p}_r))_{K^1}, \ldots, (\tilde{\mathbf{N}}_r\, (\,\mathbf{p}_r))_{K^{n_{rw}}}\right]^T \in \mathbb{R}^{n_{rw}},$$

and thus

$$\mathscr{P}^T\, \mathbf{N}_r\, (\tilde{\varphi}\mathbf{p}_r) = \left[(\tilde{\mathbf{N}}_r\, (\,\mathbf{p}_r))_{K^{\wp_1}}, \ldots, (\tilde{\mathbf{N}}_r\, (\,\mathbf{p}_r))_{K^{\wp_{m_{pw}}}}\right]^T \in \mathbb{R}^{m_{pw}}.$$

Equation (29.12) coupled with the well perforations interface conditions for all $\sigma \in \mathscr{P}_w$

$$\frac{\rho(p_{r,\sigma})}{\mu} PI^s_\sigma (\tilde{\varphi}_{K^r_\sigma} \mathbf{p}_r - p_{r,\sigma}) = \frac{\rho(p_{\text{bhp}})}{\mu} F^w_{K,\sigma}(\mathbf{P}_{w,K^w_\sigma}, p_{\text{bhp}}). \tag{29.14}$$

Let us set

$$\mathbf{p}_{r,\mathscr{P}_w} = \Big(p_{r,\sigma}, \sigma \in \mathscr{P}_w\Big),$$

using these notations, we can rewrite the reservoir subproblem (29.12)–(29.14) as follows

$$\begin{cases} \mathscr{R}_r\Big(\mathbf{p}_r, \mathbf{p}_{r,\mathscr{P}_w}\Big) = 0, \\ \mathscr{B}_{Q_T}\Big(\mathbf{p}_r, \mathbf{p}_{r,\mathscr{P}_w}\Big) = \mathscr{B}_{Q_T}\Big(\mathbf{P}_w, p_{\text{bhp}}\Big), \end{cases}$$

where $\mathscr{R}_r$ denotes the system of reservoir conservation equation, and $\mathscr{B}_{Q_T}$ denotes the total flux boundary conditions at the well perforations $\mathscr{P}_w$. Similarly, let us set

$$\mathbf{P}_{w,\mathscr{F}_{rw}} = \Big(\mathbf{P}_{w,\sigma}, \sigma \in \mathscr{F}_{rw}\Big) \text{ and } \mathbf{P}_{r,\mathscr{F}_{rw}} = \Big(\tilde{\varphi}_L \mathbf{p}_r, \sigma = K|L, \sigma \in \mathscr{F}_{rw}\Big),$$

Similarly, we can rewrite the near-well subproblem (29.7)–(29.8) as follows

$$\begin{cases} \mathscr{R}_w\Big(\mathbf{P}_w, \mathbf{P}_{w,\mathscr{F}_{rw}}\Big) = 0, \\ \mathscr{B}_{\text{robin}}\Big(\mathbf{P}_w, \mathbf{P}_{w,\mathscr{F}_{rw}}\Big) = \mathscr{B}_{\text{robin}}\Big(\mathbf{P}_r, \mathbf{P}_{r,\mathscr{F}_{rw}}\Big), \end{cases}$$

where $\mathscr{R}_w$ denotes the system of reservoir conservation equation, and $\mathscr{B}_{\text{robin}}$ denotes the Robin boundary condition for the pressure at the interface Γ_{rw}. Then, the MOR-DDM algorithm, at a given time step t^n, is the following multiplicative Schwarz algorithm which computes the reservoir and near-well solutions $\mathbf{p}_r$, and $\mathbf{P}_w$ of the coupled systems (29.12)–(29.14)–(29.7)–(29.8) solving successively the following subproblems

$$\begin{cases} \mathscr{R}_r\Big(\mathbf{p}^k_r, \mathbf{p}^k_{r,\mathscr{P}_w}\Big) = 0, \\ \mathscr{B}_{Q_T}\Big(\mathbf{p}^k_r, \mathbf{p}^k_{r,\mathscr{P}_w}\Big) = \mathscr{B}_{Q_T}\Big(\mathbf{P}^{k-1}_w, p_{\text{bhp}}\Big) \end{cases}$$

$$\begin{cases} \mathscr{R}_w\Big(\mathbf{P}^k_w, \mathbf{P}^k_{w,\mathscr{F}_{rw}}\Big) = 0, \\ \mathscr{B}_{\text{robin}}\Big(\mathbf{P}^k_w, \mathbf{P}^k_{w,\mathscr{F}_{rw}}\Big) = \mathscr{B}_{\text{robin}}\Big(\mathbf{p}^k_r, \mathbf{P}^k_{r,\mathscr{F}_{rw}}\Big) \end{cases}$$

for $k \geq 1$ until the following stopping criteria is fulfilled:

$$dQ = \frac{|\mathscr{B}_{Q_T}\left(\mathbf{P}_w^k, p_{\text{bhp}}\right) - \mathscr{B}_{Q_T}\left(\mathbf{P}_w^{k-1}, p_{\text{bhp}}\right)|}{|\mathscr{B}_{Q_T}\left(\mathbf{P}_w^k, p_{\text{bhp}}\right)|} \leq \varepsilon, \tag{29.15}$$

for a given ε.

29.3.3 Local Time Stepping

Let $t^0, \cdots, t^N$ denote the coarse time discretization on the reservoir domain with the coarse time stepping $\Delta t^n = t^n - t^{n-1} > 0$, $n = 1, \cdots, N$. Each time interval (t^{n-1}, t^n) is discretized using a local time stepping scheme in the near-well subdomain denoted by $t^{n,m}$, $m = 0, \cdots, N_n$ with $\Delta t^{n,m} = t^{n,m} - t^{n,m-1} > 0$ for all $m = 1, \cdots, N_n$, and $t^{n,0} = t^{n-1}$, $t^{n,N_n} = t^n$. Firstly, the boundary conditions at the near-well reservoir interface are interpolated in time between the two successive coarse times t^{n-1} and t^n:

$$\left\{ \begin{array}{l} \mathscr{B}_{\text{robin}}\left(\mathbf{P}_w^{n,m,k}, \mathbf{P}_{w,\mathscr{F}_{rw}}^{n,m,k}\right) = \frac{t^{n,m}-t^{n-1}}{\Delta t^n} \, \mathscr{B}_{\text{robin}}\left(\mathbf{p}_r^{k,n}, \mathbf{P}_{r,\mathscr{F}_{rw}}^{k,n}\right) \\ \qquad\qquad\qquad\qquad\qquad\quad + \frac{t^n-t^{n,m}}{\Delta t^n} \, \mathscr{B}_{\text{robin}}\left(\mathbf{p}_r^{k,n-1}, \mathbf{P}_{r,\mathscr{F}_{rw}}^{k,n-1}\right). \end{array} \right.$$

Secondly, at each well perforation of the reservoir coarse mesh, the time average of the total flux between t^{n-1} and t^n is imposed:

$$\mathscr{B}_{Q_T}\left(\mathbf{p}_r^{n,k}, \mathbf{p}_{r,\mathscr{P}_w}^{n,k}\right) = \sum_{m=1}^{N_n} \frac{\Delta t^{n,m}}{\Delta t^n} \mathscr{B}_{Q_T}\left(\mathbf{P}_w^{n,m,k-1}, p_{\text{bhp}}^{n,m}\right).$$

To construct the reduced basis for the pressure and the nonlinear term at the offline stage, we collect the snapshots at each coarse time step of the full global fine time step resolution in the LGR mesh $\mathscr{M}^{\text{lgr}}$.

29.4 Numerical Tests

The reservoir, defined by the two-dimensional domain $\Omega_r = (-L, L) \times (-L, L)$ with $L = 2.5$ km, is assumed to be heterogeneous with porosity ϕ and permeability $\mathbf{K}$ shown in Figs. 29.3 and 29.4 (SPE10, top layer). The reservoir is initially saturated with liquid (gas) at initial pressure $p_{\text{init}} = 40 \cdot 10^5$ Pa. The bottom hole pressure $p_{bhp}(t)$ at offline stage and online stage are depicted in Fig. 29.5. The vertical well injector of radius $r_w = 0.12$ m is located at the center of the reservoir. The gas mass

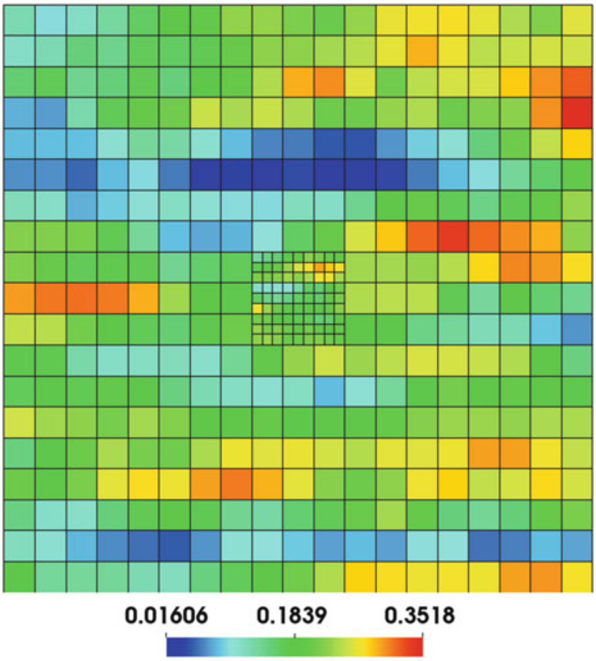


Fig. 29.3 Porosity (SPE10)

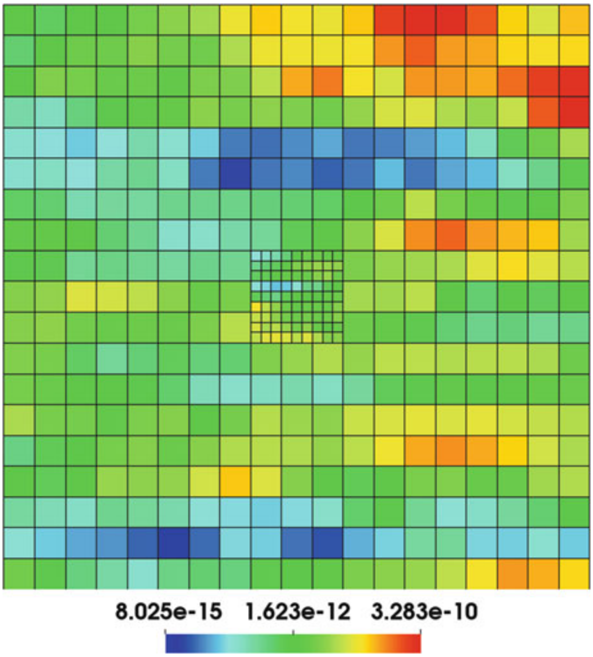


Fig. 29.4 Permeability (SPE10)

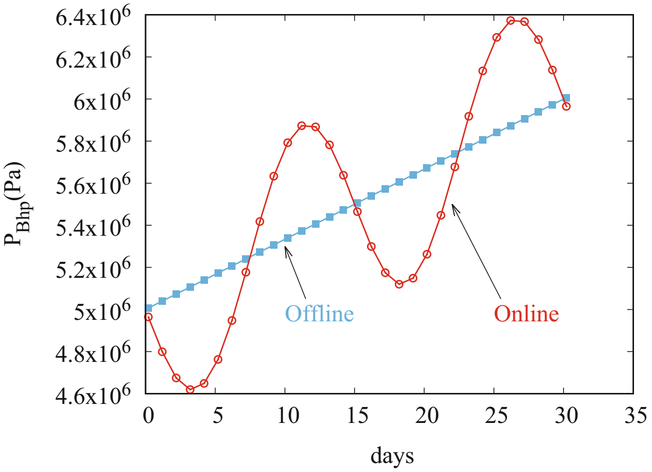


Fig. 29.5 Given bottom hole pressure (p_{bhp})

density defined by

$$\rho(p) = \frac{M}{RT}p,$$

where $R = 8.314\,\mathrm{J\,K^{-1}\,mol^{-1}}$, the molar mass $M = 0.016$ Kg, the fixed temperature $T = 323$ K. The gas viscosity is fixed to $\mu = 13 \cdot 10^{-6}$. The reservoir coarse mesh $\mathcal{M}_r$ is the uniform Cartesian mesh $M_r \times M_r$ with $M_r = 19$ of step $\Delta x = \frac{2L}{M_r} = 263.15$ m. The near-well subdomain is defined by $\Omega_w = (-L_w, L_w) \times (-L_w, L_w)$ with $L_w = 657$ m, and its mesh $\mathcal{M}_w$ is obtained, starting from the restriction of the coarse mesh $\mathcal{M}_r$ to Ω_w, by subdivision of all coarse cells in the subdomain $(-L_w + \Delta x, L_w - \Delta x) \times (-L_w + \Delta x, L_w - \Delta x)$ by a factor 3 in each direction leading to nine square fine cells per coarse cell.

In order to construct our algorithm MOR-DDM, we solve the model (29.1)–(29.2) on LGR mesh $\mathcal{M}^{\mathrm{lgr}}$ as shown in Fig. 29.2 for 30 days using the coarse time step $\Delta t = 1$ day, and a fine time stepping obtained by subdivision of each coarse time step into five sub time steps and saved the snapshots of pressure and the nonlinear term at each coarse time step. Thus, we have 30 snapshots for the both pressure and the nonlinear term. Let us denote by MOR^{n_p,m_p}-DDM, the reduced order model-domain decomposition algorithm obtained with n_p and m_p modes successively for the pressure and the nonlinear term. The solutions obtained by the MOR^{n_p,m_p}-DDM algorithm are compared in term of accuracy and CPU time to both the solution obtained with DDM algorithm and to the reference solution obtained on the LGR mesh $\mathcal{M}^{\mathrm{lgr}}$ computed with the fine time stepping.

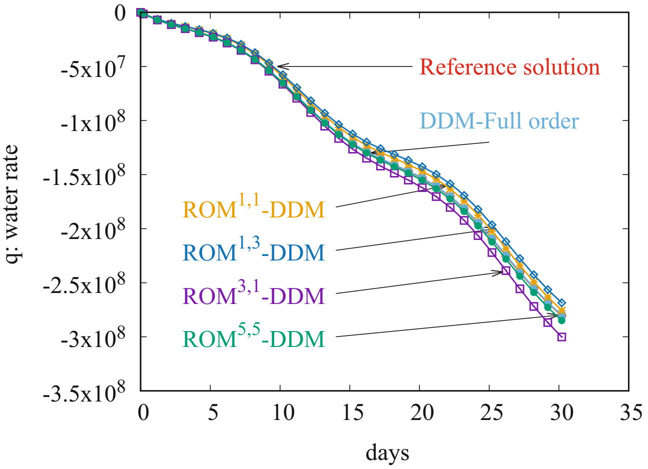


Fig. 29.6 Cumulative gas rate obtained at 30 days with different algorithms

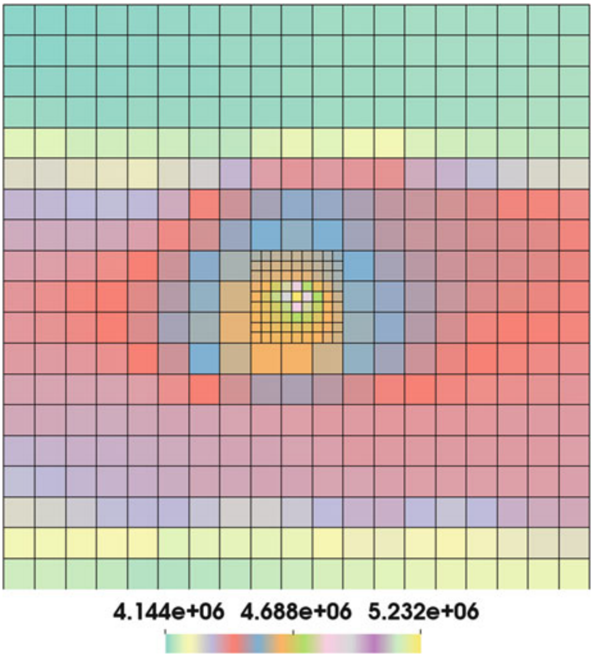


Fig. 29.7 Pressure obtained at 30 days with fine discretization using full order

The well cumulative gas flow rate as a function of time obtained with different algorithms $((n_p, m_p) \in \{1, 3, 5\}^2)$ is exhibited in Fig. 29.6. We show in Figs. 29.7 and 29.8 the pressure solution obtained at final time, successively for the reference solution and for the MOR^{5,5}-DDM algorithm. The figure shows that the solutions

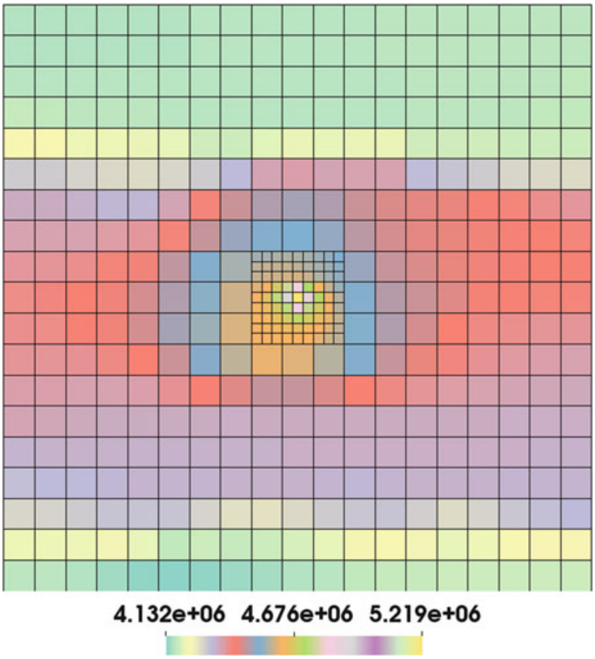


Fig. 29.8 Pressure obtained at 30 days with MOR^{5,5}-DDM

converge to the reference solution on the LGR mesh with fine time stepping as the number of modes increase

The convergence of the DDM iterations exhibited in Figs. 29.9 and 29.10 successively for DDM algorithm and for MOR^{n_p,m_p}-DDM algorithm ($(n_p, m_p) \in \{1, 3, 5\}^2$) is obtained in 2 iterations in both cases for the stopping criteria $\varepsilon = 10^{-2}$ on the relative well total flux maximum variation (29.15).

We finally give in Table 29.1, the CPU times and the relative pressure error obtained with the reference LGR algorithm using the global fine time step, the DDM algorithm and the MOR^{n_p,m_p}-DDM algorithms ($(n_p, m_p) \in \{1, 3, 5\}^2$).

These results show a factor of roughly 2 of gain in CPU time obtained with DDM algorithm with an error equal to 0.16%. A factor of almost 4 of gain in CPU time is obtained with our new algorithms MOR-DDM. This gains do not include the snapshot generating offline cost and it seems to be not significant compared to what usually obtained via MOR. However this disadvantage disappear when we apply our DD-ROM algorithm in real case of reservoir simulation application. First, the basis function generated at the offline stage will be re-used for many different input and output data, therefore the cost of our algorithm will be reduced too much compared to the re-used of the high fidelity model.

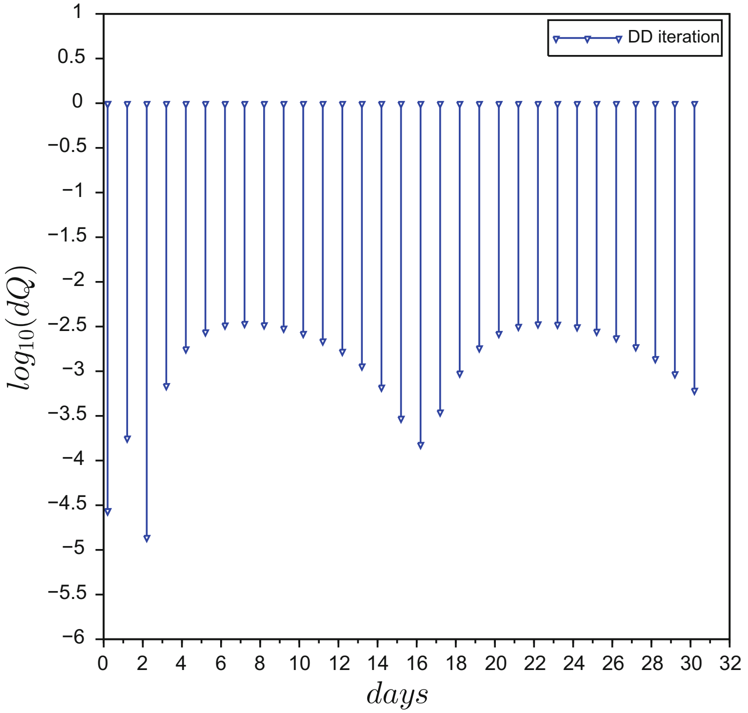


Fig. 29.9 Convergence of *dQ* obtained by DDM algorithm using full order

Second, due to the use of a full order on the near-well region, the CPU gains will increase whenever the size of the near-well region decreasing compared to the rest of the reservoir. In our example the near well region is almost one fifth of the reservoir domain whereas in real case the near well region is limited to few meters and the reservoir stretches mostly over several tens of kilometers.

The error obtained with the different modes number is close to the error obtained with DDM algorithm using full order.

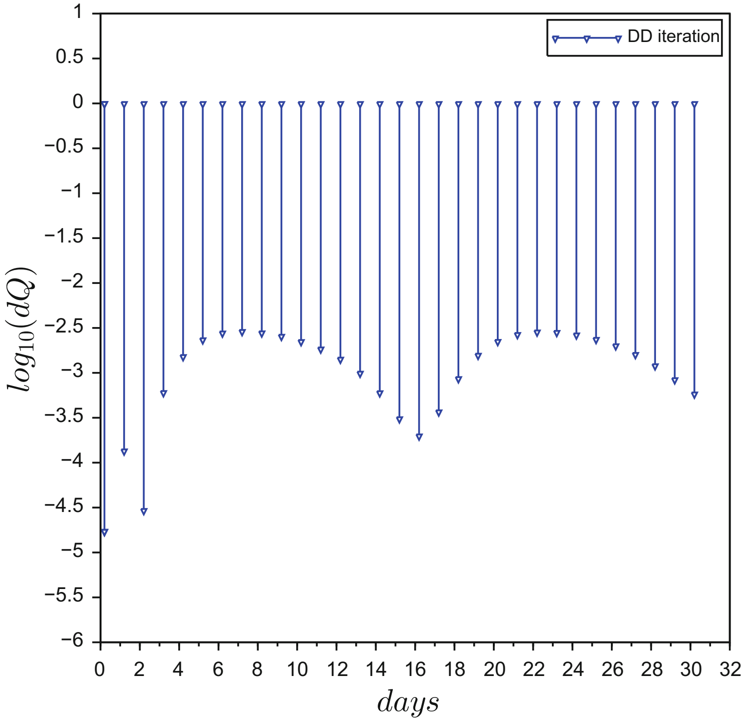


Fig. 29.10 Convergence of dQ obtained by MOR^{3,3}-DDM algorithm

Table 29.1 Relative pressure error and CPU time obtained with different algorithms

Methods	CPU time (s)	Pressure error
LGR-Fine time step-Full order[a]	86	–
DDM-Full order	40	0.0016
MOR^{1,1}-DDM	18	0.0102
MOR^{3,1}-DDM	20	0.0177
MOR^{1,3}-DDM	20	0.0154
MOR^{3,3}-DDM	21	0.0049
MOR^{5,3}-DDM	20	0.0050
MOR^{3,5}-DDM	20	0.0050
MOR^{5,5}-DDM	20	0.0044

[a] Reference solution

29.5 Conclusion

A model order reduction algorithm for a compressible flow model in porous media coupling near-well regions locally refined in space and time with a coarser reservoir discretization has been presented. The algorithm is based on domain decomposition method and using POD locally for the pressure and DEIM locally for the nonlinear term. The algorithm has been implemented in 2D for a gas flow through porous media in a heterogeneous reservoir with an injection well. The numerical results show good behavior of our algorithm that provides good accuracy compared to the reference solution obtained with full order on LGR using a global fine time step. Furthermore we observe important gains in CPU time for a cost approximately 4 times less. Compared with the solution obtained with full order using local time step (DDM algorithm) we get a CPU time savings for a cost approximately 2 times less.

Acknowledgements This publication was made possible by NPRP award [NPRP 7-1482-1-278] from the Qatar National Research Fund (a member of The Qatar Foundation). Additionally, this project was partially supported by the European Union's Horizon 2020, research and innovation programme under the Marie Sklodowska-Curie grant agreement N 644202.

References

1. Afra, S., Gildin, E., Tarrahi, M.: Heterogeneous reservoir characterization using efficient parameterization through higher order svd (hosvd). In: American Control Conference. IEEE, Portland, OR (2014)
2. Antil, H., Heinkenschloss, M., Hoppe, R.H.W., Sorensen, D.C.: Domain decomposition and model reduction for the numerical solution of PDE constrained optimization problems with localized optimization variables. Comput. Vis. Sci. **13**(6), 249–264 (2010)
3. Antoulas, A., Sorensen, D., Gugercin, S.: A survey of model reduction methods for large-scale systems. Contemp. Math. Numer. Algorithms **280**, 193–220 (2001)
4. Baiges, J., Codina, R., Idelsohn, S.: A domain decomposition strategy for reduced order models. Application to the Incompressible Navier-Stokes Equations. Comput. Methods Appl. Mech. Eng. **267**, 23–42 (2013)
5. Buffoni M., Telib, H., Lollo, A.: Iterative methods for model reduction by domain decomposition. Comput. Fluids **38**(6), 1160–1167 (2009)
6. Cardoso, M., Durlofsky, L.: Use of reduced-order modeling procedures for production optimization. SPE J. **15**(2), 426–435 (2010)
7. Chaturantabut, S., Sorensen, D.C.: Nonlinear model reduction via discrete empirical interpolation. SIAM J. Sci. Comput. **32**(5), 2737–2764 (2010). doi:10.1137/090766498
8. Corigliano, A., Dossi, M., Mariani, S.: Domain decomposition and model order reduction methods applied to the simulation of multi-physics problems in MEMS. Comput. Struct. **122**, 113–127 (2013)
9. Corigliano, A., Dossi, M., Mariani, S.: Model order reduction and domain decomposition strategies for the solution of the dynamic elastic-plastic structural problem. Comput. Methods Appl. Mech. Eng. **290**, 127–155 (2015)
10. Doren, J., Markovinovic R., Jansen, J.-D.: Reduced-order optimal control of water flooding using proper orthogonal decomposition. Comput. Geosci. **10**(1), 137–158 (2006). doi:10.1007/s10596-005-9014-2. http://dx.doi.org/10.1007/s10596-005-9014-2
11. Doren, J.V., Markovinovic, R., Cansen, J.: Reduced-order optimal control of waterflooding using pod. In: 9th European Conference of the Mathematics of Oil Recovery. EAGE, Cannes (2004)

12. Efendiev, Y., Romanovskay, A., Gildin, E., Ghasemi, M.: Nonlinear complexity reduction for fast simulation of flow in heterogeneous porous media. In: SPE Reservoir Simulation Symposium. Society of Petroleum Engineers, The Woodlands, TX. SPE 163618-MS (2013). http://dx.doi.org/10.2118/163618-MS
13. Ewing, R.E., Lazarov, R.D., Vassilevski, P.S.: Finite difference schemes on grids with local refinement on time and space for parabolic problems. Derivation, stability and error analysis. Computing **45**, 193–215 (1990)
14. Eymard, R., Gallouët, T., Herbin, R.: Finite volume methods. Handb. Numer. Anal. **7**, 713–1018 (2000)
15. Gander, M.J.: Optimized Schwarz methods. SIAM J. Numer. Anal. **44**, 699–731 (2006)
16. Ghasemi, M., Zhao, S., Insperger, T., Kalmar-Nagy, T.: Act-and-wait control of discrete systems with random delays. In: American Control Conference (ACC), pp. 5440–5443. IEEE, Montreal (2012). http://dx.doi.org/10.1109/ACC.2012.6315674
17. Ghasemi, M., Ashraf, I., Gildin, E.: Reduced order modeling in reservoir simulation using the bilinear approximaion techniques. In: SPE Latin American and Caribbean Petroleum Engineering Conference. Society of Petroleum Engineers, Maracaibo. SPE 169357-MS (2014). http://dx.doi.org/10.2118/169357-MS
18. Ghasemi M., Yang, Y., Gildin, E., Efendiev Y., Calo, V.: Fast multiscale reservoir simulations using POD-DEIM model reduction. In: SPE Reservoir Simulation Symposium. Houston, TX, pp. 23–25 (2015)
19. Ghommem, M., Calo, V.M., Efendiev, Y., Gildin, E.: Complexity reduction of multi-phase flows in heterogeneous porous media. In: SPE Kuwait Oil and Gas Show and Conference. SPE, Kuwait City. SPE 167295 (2013)
20. Gildin, E., Ghasemi, M.: A new model reduction technique applied to reservoir simulation. In: 14th European conference on the mathematics of oil recovery. European Association of Geoscientists and Engineers, Sicily (2014). http://dx.doi.org/10.3997/2214-4609.20141820
21. Gildin, E., Lopez, T.J.: Closed-loop reservoir management: do we need complex models. In: SPE Digital Energy Conference and Exhibition. The Woodlands, TX (2011)
22. Heijn, T., Markovinovic, R., Jansen, J.: Generation of low-order reservoir models using system-theoretical concepts. SPE J. **9**(2) (2004)
23. Jafarpour, B., Tarrahi, M.: Assessing the performance of the ensemble kalman filter for subsurface flow data integration under variogram uncertainty. Water Resour. Res. **47**(5) (2011)
24. Kheriji, W., Masson, R., Moncorgé, A.: Nearwell local space and time refinement in reservoir simulation. Math. Comput. Simul. **118**, 273–292 (2015)
25. Lerlertpakdee, P., Jafarpour, B., Gildin, E.: Efficient production optimization with flow-network models. SPE J. **19**, 1–83 (2014)
26. Mlacnik, M.J.: Using well windows in full field reservoir simulations. Ph.D. Thesis, University of Leoben (2002)
27. Oliver, D.S., Reynolds, A.C., Liu, N.: Inverse Theory for Petroleum Reservoir Characterization and History Matching, vol. 1. Cambridge University Press, Cambridge (2008)
28. Peaceman, D.W.: Fundamentals of Numerical Reservoir Simulations. Elsevier, Amsterdam (1977)
29. Queipo, N.V., Pintos, S., Rincón, N., Contreras, N., Colmenares, J.: Surrogate modeling-based optimization for the integration of static and dynamic data into a reservoir description. J. Pet. Sci. Eng. **35**(3), 167–181 (2002)
30. Sun, K., Glowinski, R., Heinkenschloss, M., Sorensen, D.C.: Domain decomposition and model reduction of systems with local nonlinearities. In: Proceedings of ENUMATH 2007. The 7th European Conference on Numerical Mathematics and Advanced Applications, Graz, pp. 389–396 (2008)
31. Voneiff, G., Sadeghi, S., Bastian, P., Wolters, B., Jochen, J., Chow, B., Gatens, M.: Probabilistic forecasting of horizontal well performance in unconventional reservoirs using publicly-available completion data. In: SPE Unconventional Resources Conference. Society of Petroleum Engineers, The Woodlands, TX (2014)

Chapter 30
Time-Dependent Parametric Model Order Reduction for Material Removal Simulations

Michael Baumann, Dominik Hamann, and Peter Eberhard

Abstract Machining of thin and lightweight structures is a crucial manufacturing step in industries ranging from aerospace to power engineering. In order to enable efficient simulations of elastic workpieces and solve typical tasks like the prediction of process stability, reduced elastic models have to be determined by model order reduction. Thereby, the system matrices need to be constant, which cannot be assumed for elastic bodies with varying geometry due to material removal. In this contribution we propose a technique to generate reduced elastic bodies for systems with time-varying geometries and their application in time-domain simulations. Therefore, the model is described as a parameter-dependent system. Due to the fact that the considered parameter varies in time-domain simulations, time-dependent parametric model order reduction techniques for elastic bodies are presented.

30.1 Introduction

In elastic multibody simulations moving loads can be implemented with parametric model order reduction. Usually the parameter dependency is actually also a time dependency, thus time-dependent parametric model order reduction has to be developed and implemented, see [11, 13]. The application of parametric model order reduction, neglecting the time variance in the modeling, does not consider the variation of the reduced system dynamics. The time-dependent parametric model order reduction eliminates the implicit simplification in modeling. However, these additional terms have been frequently and consciously neglected in many applications of moving loads, due to their minor influence in these applications. In [11, 13] time-dependent parametric model order reduction is investigated for moving load problems, where the finite element mesh remains constant. The parameter

M. Baumann • D. Hamann • P. Eberhard (✉)
Institute of Engineering and Computational Mechanics, University of Stuttgart,
Pfaffenwaldring 9, 70569 Stuttgart, Germany
e-mail: dominik.hamann@itm.uni-stuttgart.de; peter.eberhard@itm.uni-stuttgart.de

P. Benner et al. (eds.), *Model Reduction of Parametrized Systems*,
MS&A 17, DOI 10.1007/978-3-319-58786-8_30

dependency of the T-shaped plate is linked to the machining progress. On the one hand the contact point moves due to the feed of the milling tool and on the other hand the geometry of the T-shaped plate is affected due to machining. Based on the milling process a distinctive change of the workpiece dynamics can be determined. Thus, the change of the eigenfrequencies is to be mentioned, [1, 5, 6]. The investigated application of this contribution significantly differs from commonly investigated applications by these characteristics. Furthermore, machining processes exhibit a strong sensitivity to varying dynamics. Thus, the simplifications in the modeling may effect the stability of the process. Hence, the model of the T-shaped plate is a remarkable example for the application of parametric model order reduction techniques and motivated by realistic applications. Practical problems include machining of frame components, milling of blades for aircraft propellers or turbines. The applications demand high geometric accuracy and an excellent surface finish. In [4], a turning process is investigated using a parametric flexible multibody model. The new contribution of this paper is the comparison of parametric model order reduction and time-dependent parametric model order reduction in material removal simulations.

The structure of the paper is as follows. First, the theoretical approach of time-dependent parametric model order reduction is presented in Sect. 30.2. Afterwards, the modeling of the milling force, Sect. 30.3, and the workpiece dynamics, Sect. 30.4, of the investigated example is stated. Section 30.5 presents the results with the investigation of the T-shaped plate in frequency domain as well as the milling process in time domain.

30.2 Time-Dependent Parametric Model Order Reduction

In this section the mathematical differences of the time dependent parametric model order reduction (PTMOR) compared to parametric model order reduction (PMOR) are presented. Starting with model order reduction (MOR) of linear time invariant systems, PMOR offers a meaningful interpolation of the systems. After describing some basics, the computational handling of the additional terms of PTMOR is shown.

The linear time invariant system from structural mechanics

$$\mathbf{M}\ddot{\mathbf{q}}(t) + \mathbf{D}\dot{\mathbf{q}}(t) + \mathbf{K}\mathbf{q}(t) = \mathbf{B}\mathbf{u}(t)\,, \tag{30.1a}$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{q}(t) \tag{30.1b}$$

is considered, which is an ordinary differential equation obtained by local discretization of partial differential equations with the finite element method. Here, the symmetric mass, damping, and stiffness matrices $\mathbf{M}$, $\mathbf{D}$, $\mathbf{K} \in \mathbb{R}^{N\times N}$ and the elastic coordinates $\mathbf{q}(t) \in \mathbb{R}^N$ are taken into account. The input matrix $\mathbf{B} \in \mathbb{R}^{N\times r}$

distributes the components of the actuating force $\mathbf{u}(t) \in \mathbb{R}^r$, whereas the output matrix $\mathbf{C} \in \mathbb{R}^{w \times N}$ specifies the favored output $\mathbf{y}(t) \in \mathbb{R}^w$.

An approximation of the high order model by MOR techniques, see [3], based on the projection

$$\mathbf{q}(t) \approx \mathbf{V}\bar{\mathbf{q}}(t) \tag{30.2}$$

with the projection matrix $\mathbf{V} \in \mathbb{R}^{N \times n}$, and $n \ll N$ yields the reduced order model

$$\underbrace{\mathbf{W}^{\mathsf{T}}\mathbf{M}\mathbf{V}}_{\bar{\mathbf{M}}}\,\ddot{\bar{\mathbf{q}}}(t) + \underbrace{\mathbf{W}^{\mathsf{T}}\mathbf{D}\mathbf{V}}_{\bar{\mathbf{D}}}\,\dot{\bar{\mathbf{q}}}(t) + \underbrace{\mathbf{W}^{\mathsf{T}}\mathbf{K}\mathbf{V}}_{\bar{\mathbf{K}}}\,\bar{\mathbf{q}}(t) = \underbrace{\mathbf{W}^{\mathsf{T}}\mathbf{B}}_{\bar{\mathbf{B}}}\,\mathbf{u}(t)\,, \tag{30.3a}$$

$$\bar{\mathbf{y}}(t) = \underbrace{\mathbf{C}\mathbf{V}}_{\bar{\mathbf{C}}}\,\bar{\mathbf{q}}(t) \tag{30.3b}$$

by means of the Petrov-Galerkin projection and the left projection matrix $\mathbf{W} \in \mathbb{R}^{N \times n}$. If for the projection matrices $\mathbf{W} = \mathbf{V}$, orthogonal projection instead of oblique projection is used, whereby the preservation of structural properties is ensured.

The representation of the interesting system dynamics by the reduced order model, depicted by the bar symbol, can save computational effort while preserving the essential dynamics and neglecting the non-essential dynamics.

If an interpolation of different reduced order models is sought, PMOR techniques can be used. The demand of interpolated systems could, e.g., be motivated by optimization problems for which the preparation of high order models for every parameter p is too expensive.

Generally the subspaces of individually reduced systems differ and, therefore, the elastic coordinates can differ as well

$$\bar{\mathbf{q}}_1(t) \neq \ldots \neq \bar{\mathbf{q}}_i \neq \ldots \neq \bar{\mathbf{q}}_k(t)\,. \tag{30.4}$$

Hence, the direct matrix interpolation is not meaningful and the matrix interpolation of different reduced order models can be prepared by the additional individual transformation

$$\bar{\mathbf{q}}_i(t) = \mathbf{T}_i\tilde{\mathbf{q}}(t) \tag{30.5}$$

with

$$\mathbf{T}_i = \left(\mathbf{R}^{\mathsf{T}}\mathbf{V}_i\right)^{-1}, \tag{30.6}$$

see [9]. In order to select a common subspace, the singular value decomposition (SVD) of the union of all subspaces

$$\mathbf{U}\Sigma\mathbf{N}^{\mathsf{T}} = \operatorname{svd}\bigl([\underbrace{\mathbf{v}_1, \ldots, \mathbf{v}_n}_{\mathbf{V}_1}, \ldots, \mathbf{v}_{nk}]\bigr) \ , \tag{30.7a}$$

$$\mathbf{U} = [\underbrace{\mathbf{u}_1, \ldots, \mathbf{u}_n}_{\mathbf{R}}, \ldots, \mathbf{u}_{nk}] \tag{30.7b}$$

can be used. Here, k reduced order systems of order n are considered. Choosing the common subspace based on an SVD is not mandatory. Another simple way is to choose the subspace of one reduced order model $\mathbf{R} = \mathbf{V}_i$. The approach yields the approximation

$$\mathbf{q}(t) \approx \underbrace{\mathbf{V}_i\mathbf{T}_i}_{\widetilde{\mathbf{V}}_i} \ \tilde{\mathbf{q}}(t) \tag{30.8}$$

and the linear parameter varying system

$$\widetilde{\mathbf{M}}(p)\ddot{\tilde{\mathbf{q}}}(t) + \widetilde{\mathbf{D}}(p)\dot{\tilde{\mathbf{q}}}(t) + \widetilde{\mathbf{K}}(p)\tilde{\mathbf{q}}(t) = \widetilde{\mathbf{B}}(p)\mathbf{u}(t) \ , \tag{30.9a}$$

$$\tilde{\mathbf{y}}(t) = \widetilde{\mathbf{C}}(p)\tilde{\mathbf{q}}(t) \ . \tag{30.9b}$$

In this paper, for simpler representation only a scalar parameter p is considered but everything can be generalized to parameter vectors. In this regard, the system dynamics is represented by the interpolated system matrices

$$\widetilde{\mathbf{N}}(p) = \sum_{i=1}^{k} c_i(p) \underbrace{\widetilde{\mathbf{V}}_i^{\mathsf{T}}\mathbf{N}_i\widetilde{\mathbf{V}}_i}_{\widetilde{\mathbf{N}}_i} \quad , \qquad \mathbf{N}_i \in \{\mathbf{M}_i,\ \mathbf{D}_i,\ \mathbf{K}_i,\ \mathbf{B}_i\ ,\mathbf{C}_i\} \tag{30.10}$$

depicted by the tilde symbol. The parameter-dependent coefficients $c_i(p)$ are chosen by the interpolation approach, e.g. using linear interpolation or cubic splines.

Various modeling problems require a time dependency of the parameter, i.e. $p = p(t)$, which is then considered by PTMOR. An application of this extension is the moving load problem. As a consequence of the time dependent parameter, there results a time dependency of the projection matrix

$$\widetilde{\mathbf{V}} = \widetilde{\mathbf{V}}\bigl(p(t)\bigr) = \widetilde{\mathbf{V}}(t) \tag{30.11}$$

which yields the approximations

$$\mathbf{q}(t) \approx \widetilde{\mathbf{V}}(t)\hat{\mathbf{q}}(t)\,, \tag{30.12a}$$

$$\dot{\mathbf{q}}(t) \approx \widetilde{\mathbf{V}}(t)\dot{\hat{\mathbf{q}}}(t) + \dot{\widetilde{\mathbf{V}}}(t)\hat{\mathbf{q}}(t)\,, \tag{30.12b}$$

$$\ddot{\mathbf{q}}(t) \approx \widetilde{\mathbf{V}}(t)\ddot{\hat{\mathbf{q}}}(t) + 2\dot{\widetilde{\mathbf{V}}}(t)\dot{\hat{\mathbf{q}}}(t) + \ddot{\widetilde{\mathbf{V}}}(t)\hat{\mathbf{q}}(t)\,, \tag{30.12c}$$

whereby the derivatives of the elastic coordinates receive additional terms due to Leibniz's rule, see [11, 13].

For the calculation of the additional terms, the time derivatives of the projection matrix $\widetilde{\mathbf{V}}(t)$ are separated corresponding to

$$\dot{\widetilde{\mathbf{V}}}(t) = \frac{\mathrm{d}\widetilde{\mathbf{V}}(t)}{\mathrm{d}t} = \frac{\partial\widetilde{\mathbf{V}}}{\partial p}\frac{\mathrm{d}p}{\mathrm{d}t} + \underbrace{\frac{\partial\widetilde{\mathbf{V}}}{\partial t}}_{\mathbf{0}} = \frac{\partial\widetilde{\mathbf{V}}}{\partial p}\dot{p}(t)\,, \tag{30.13a}$$

$$\ddot{\widetilde{\mathbf{V}}}(t) = \frac{\partial^2\widetilde{\mathbf{V}}}{\partial p^2}\dot{p}^2(t) + \frac{\partial\widetilde{\mathbf{V}}}{\partial p}\ddot{p}(t) \tag{30.13b}$$

into parameter dependent matrices and time dependent parameter derivations. The time dependent matrices vanish since there is no direct time dependency of the projection matrix.

The linear time varying system

$$\Big(\underbrace{\widetilde{\mathbf{M}}_0(p)}_{\widehat{\mathbf{M}}}\Big)\ddot{\hat{\mathbf{q}}}(t) + \Big(\underbrace{\widetilde{\mathbf{D}}_0(p) + 2\widetilde{\mathbf{M}}_1(p)\dot{p}}_{\widehat{\mathbf{D}}}\Big)\dot{\hat{\mathbf{q}}}(t)$$
$$+\Big(\underbrace{\widetilde{\mathbf{K}}_0(p) + \widetilde{\mathbf{D}}_1(p)\dot{p} + \widetilde{\mathbf{M}}_2(p)\dot{p}^2 + \widetilde{\mathbf{M}}_1(p)\ddot{p}}_{\widehat{\mathbf{K}}}\Big)\hat{\mathbf{q}}(t) = \underbrace{\widetilde{\mathbf{B}}_0(p)}_{\widehat{\mathbf{B}}}\,\mathbf{u}(t)\,, \tag{30.14a}$$

$$\hat{\mathbf{y}}(t) = \underbrace{\widetilde{\mathbf{C}}_0(p)}_{\widehat{\mathbf{C}}}\,\hat{\mathbf{q}}(t) \tag{30.14b}$$

is obtained with

$$\widetilde{\mathbf{N}}_r(p) = \sum_{i=1}^{n} c_i(p)\,\underbrace{\widetilde{\mathbf{V}}_i^{\mathrm{T}}\mathbf{N}_i\frac{\partial^r\widetilde{\mathbf{V}}_i}{\partial p^r}}_{\widetilde{\mathbf{N}}_{i,r}}\,. \tag{30.15}$$

All matrices of this linear time varying system can be calculated in an offline step whereas just the interpolation has to be done online. The matrices $\widetilde{\mathbf{N}}_r$ with $r = 0$ match to the matrices of the PMOR approach. The matrices which are affected by partial derivatives $\widetilde{\mathbf{N}}_r$ with $r > 0$ are the new terms in the PTMOR approach distinguishing it from the PMOR approach, however they are not available directly from the high order model. Thus, numerical methods, for example difference quotients, have to be used. Another possibility is in the case of polynomial interpolation, such as cubic spline interpolation, utilizing derivatives of these polynomials.

30.3 Modeling of the Milling Force

Milling is described by a common force law in the stability analysis of machining processes, [8]. The centerpiece of this approach is the dynamic chip thickness approximation which uses an approximation of the nominal chip thickness and one based on the relative displacement of the workpiece and the tool. Both the current displacement and the material removal one rotation before, is an indication of the workpiece surface and thus the chip thickness. The current displacement, which indicates the current machined surface, is taken into account in the next rotation. This process is not necessarily stable and can lead to self-exited vibrations, called chatter, [12]. In stability analysis of machining processes, the dynamics of the workpiece is often modeled in a rather simple way, e.g. by single-degree-of-freedom oscillators. In contrast, high dimensional finite element models are used for the modeling of chipping, where the finite element mesh has to be adapted very frequently. The use of a (time-dependent) parametric reduced-order model represents an approach to combine the advantages of both existing approaches. On the one hand, we have a small model, which enables complex stability analysis, and on the other hand, we preserve the dynamics of the workpiece as detailed as necessary.

Figure 30.1 illustrates the mechanical model of up-milling. The sketch shows a workpiece and a symmetric milling tool with $M = 2$ teeth. Here, only the displacement of the workpiece $\mathbf{y}(t) = [x(t) \;\; y(t)]^\mathsf{T}$ is taken into account. The dynamic chip

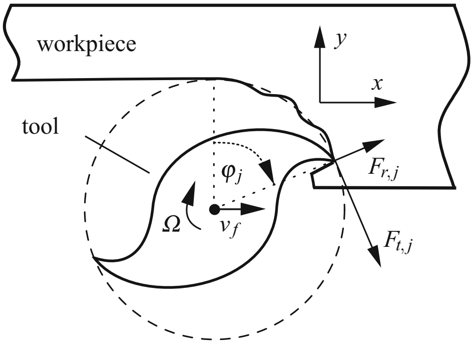


Fig. 30.1 Mechanical model of material removal by milling

thickness of the tooth j

$$h_j = g(\varphi_j)\begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}\cos(\varphi_j) & -\sin(\varphi_j)\\ \sin(\varphi_j) & \cos(\varphi_j)\end{bmatrix}\left(\begin{bmatrix}f_z\\ 0\end{bmatrix} - \mathbf{y}(t) + \mathbf{y}(t-\tau)\right) \tag{30.16}$$

is calculated using the current displacement $\mathbf{y}(t)$ and the one of the rotations before $\mathbf{y}(t-\tau)$, where the time delay $\tau = 60/(M\Omega)$ depends on the revolution speed Ω and, in case of symmetry, of the number of teeth M. With the feed per tooth $f_z = 0.1 \cdot 10^{-3}$ m, the calculation of the radial part of the chip thickness and the screen function

$$g(\varphi_j) = \begin{cases}1 & \text{if } \phi_{\text{en}} < \operatorname{mod}(\varphi_j, 2\pi) < \phi_{\text{ex}}\\ 0 & \text{otherwise,}\end{cases} \tag{30.17}$$

the dynamic chip thickness calculation is almost complete. The screen function considers that either the tooth j is currently cutting or not, which is detected using the modulo operator $\operatorname{mod}(a, b)$ returning the remainder a/b, the entry angle ϕ_{en}, and the exit angle ϕ_{ex}. For up-milling $\phi_{\text{en}} = 0$, as illustrated, and ϕ_{ex} is calculated including the radial immersion and the diameter of the tool. A nonsensical, negative chip thickness is avoided by

$$\bar{h}_j = \max(h_j, 0)\,. \tag{30.18}$$

The resulting forces in the tangential and radial direction at the tooth j are

$$\mathbf{F}_{tr,j} = \begin{bmatrix}F_{t,j}\\ F_{r,j}\end{bmatrix} = a_p\begin{bmatrix}k_t\bar{h}_j^{q_t}\\ k_r\bar{h}_j^{q_r}\end{bmatrix} \tag{30.19}$$

with the axial immersion a_p, cutting force coefficient $k_t = 107 \cdot 10^6\,\text{N/m}^{1+\text{q}_\text{t}}$ and $k_r = 40 \cdot 10^6\,\text{N/m}^{1+\text{q}_\text{r}}$, and exponent $q_{t,r} = 0.75$ in the tangential and radial direction. The tangential and radial forces at the teeth have to be transformed in x and y direction

$$\mathbf{F}_{xy,j} = \begin{bmatrix}\cos(\varphi_j) & \sin(\varphi_j)\\ -\sin(\varphi_j) & \cos(\varphi_j)\end{bmatrix}\mathbf{F}_{tr,j} \tag{30.20}$$

and summed up

$$\mathbf{F}_{xy}\big(t, \mathbf{y}(t), \mathbf{y}(t-\tau)\big) = \sum_{j=1}^{M}\mathbf{F}_{xy,j}\,. \tag{30.21}$$

The presented mathematical model of the chipping force underlies several simplifications which enable a simple and powerful linearization for stability analysis. For instance the unperturbed chip thickness is a circular approximation of the trochoidal geometry or the neglect of the tangential displacement are to be mentioned.

30.4 Modeling of the Workpiece Dynamics

For the calculation of the varying workpiece dynamics, an efficient implementation is available. The geometry of the workpiece, represented by the mesh, is adapted to several states of the machining progress. Nodes are shifted in the y direction, corresponding to the ideal tool path, and the relevant elements are updated, see Fig. 30.2.

The model of the T-shaped plate has primary dimensions of $0.17\,\mathrm{m} \times 0.03\,\mathrm{m} \times 0.15\,\mathrm{m}$ and is set up by plates with the thickness $0.01\,\mathrm{m}$. The tool with a diameter of $0.02\,\mathrm{m}$ machines a notch with a cross section of $0.005\,\mathrm{m} \times 0.01\,\mathrm{m}$. The axial immersion $a_p = 0.01\,\mathrm{m}$ is constant while the milling tool moves $0.02\,\mathrm{m}$ in the x direction. Linear elements with a feed size of $1/3 \cdot 10^{-3}\,\mathrm{m}$ are used. This yields $30,888$ degrees of freedom for the constrained model due to the fixed bottom of the workpiece.

The mass matrix $\mathbf{M}$ and the stiffness matrix $\mathbf{K}$ are calculated for steel with density $\rho = 7.8 \cdot 10^3\,\mathrm{kg/m^3}$, Young's modulus $E = 210 \cdot 10^9\,\mathrm{N/m^2}$ and Poisson ratio $\nu = 0.3$. For the calculation of the damping matrix $\mathbf{D}$, Rayleigh damping $\mathbf{D} = \alpha\mathbf{M} + \beta\mathbf{K}$ is used. The mass proportional factor is $\alpha = 3.86\,\mathrm{s}^{-1}$ and the stiffness proportional factor is $\beta = 2.25 \cdot 10^{-5}\,\mathrm{s}$.

To accommodate the presented chipping force the input and output matrix are chosen to the x and y degrees of freedom of one node, thus $\mathbf{C}^\mathsf{T} = \mathbf{B}$ and $r = w = 2$. The chosen surface node is at the same x position as the milling tool and

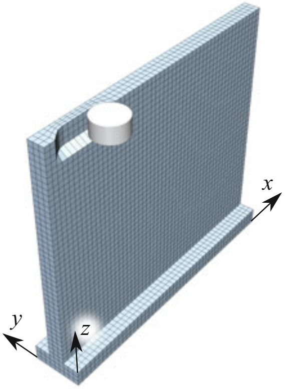


Fig. 30.2 T-shaped workpiece

in the middle of the axial immersion. Generally, there are many nodes available and complex material removal algorithms can be used for the implementation of the load application.

The workpiece dynamics is approximated with projection matrices determined by modal truncation, component mode synthesis and moment matching, see [2, 3, 7]. The order of the reduced systems is chosen to $n = 10$, thus $m \in [1,\ n]$. First, modal truncation is realized by calculating eigenmodes of the conservative system

$$\left(\lambda_m^2 \mathbf{M} + \mathbf{K}\right) \mathit{\Phi}_{\mathrm{eig},m} = \mathbf{0} \,. \tag{30.22}$$

In order to include static accuracy

$$\mathbf{K}\mathit{\Phi}_{\mathrm{stat}} = \mathbf{B} \tag{30.23}$$

with $\mathit{\Phi}_{\mathrm{stat}} \in \mathbb{R}^{30\,888 \times 2}$ it follows

$$\mathbf{V}_{\mathrm{cms}} = \left[\mathit{\Phi}_{\mathrm{eig},1},\ \ldots,\ \mathit{\Phi}_{\mathrm{eig},8},\ \mathit{\Phi}_{\mathrm{stat}}\right] \,. \tag{30.24}$$

In case of MOR with moment matching the Krylov subspace is calculated with an Arnoldi algorithm [10]. Here, $\mathbf{W}_{\mathrm{kry}} = \mathbf{V}_{\mathrm{kry}}$ due to $\mathbf{C}^{\mathsf{T}} = \mathbf{B}$ and five moments are matched at frequency $f_{\mathrm{kry}} = 450\,\mathrm{Hz}$, next to the first eigenfrequency of the T-shaped plate.

30.5 Results

We investigate the additional terms of PTMOR which are usually neglected in the PMOR approach using the example of the milling process. Hence, investigations in the frequency domain and time domain are considered. In the time domain, the entire process is simulated with standard integration schemes. In the frequency domain, the frozen transfer function is investigated, [14].

30.5.1 Frequency Domain

The investigation of the dynamics of the presented nonlinear, time varying and delayed system with methods of linear time invariant systems is motivated by different characteristics. Firstly, the variance of the investigated system is quite slow in simulations with realistic milling parameters. Secondly, the actuating force $\mathbf{u}(t) = \mathbf{F}_{xy}\big(t, \mathbf{y}(t), \mathbf{y}(t-\tau)\big)$ is almost periodic, due to the periodicity of the coefficients despite the state and delayed state dependency considering stable machining processes, [5, 8]. The oscillations of stable milling processes only contain harmonics of the tooth passing frequency $f_t = N\mathit{\Omega}/60$.

The transfer function which represents the response of linear time invariant systems reads

$$\mathbf{H}(s) = \mathbf{C}\left(s^2\mathbf{M} + s\mathbf{D} + \mathbf{K}\right)^{-1}\mathbf{B} \tag{30.25}$$

and is obtained with the Laplace transformation with the complex frequency parameter $s = \imath\omega$, the angular frequency $\omega = 2\pi f$, the imaginary unit $\imath$ and the frequency f of the actuation. Considering the time dependency due to the parameter change, the frozen transfer function at one time point is given by $\widehat{\mathbf{H}}(s,t) = \widehat{\mathbf{H}}(s,p,\varOmega)$, where the current machining progress is represented by the parameter p and the velocity of the variation depends on the revolution speed $\varOmega$ due to the feed velocity $v_f = f_z N\varOmega/60$.

Using a Frobenius error measure, the absolute error is defined by

$$\varepsilon^{\text{abs}}_{\varTheta,\varXi}(s) = \|\varTheta(s) - \varXi(s)\|_{\mathcal{F}}\ , \tag{30.26}$$

where $\varTheta(s)$ and $\varXi(s)$ are representing transfer functions, of the original $\mathbf{H}(s)$ and the reduced system $\bar{\mathbf{H}}(s)$, and the Frobenius norm $\|\cdot\|_{\mathcal{F}}$ is used.

An illustration of the transfer function of the high order model and of absolute errors is shown in Fig. 30.3. As a result, the magnitude of the influence of the PTMOR approach compared to the absolute reduction error is small for this simulation scenario. The absolute reduction error $\|\widetilde{\mathbf{H}} - \mathbf{H}\|_{\mathcal{F}}$ of the component mode synthesis shows a high dependency on the frequency. The attachment of static modes yields a static correctness whereas the dynamic error is about 10^{-10}. The absolute deviation of the PTMOR to the PMOR approach $\|\widehat{\mathbf{H}} - \widetilde{\mathbf{H}}\|_{\mathcal{F}}$ is about 10^{-13}, however, the transfer

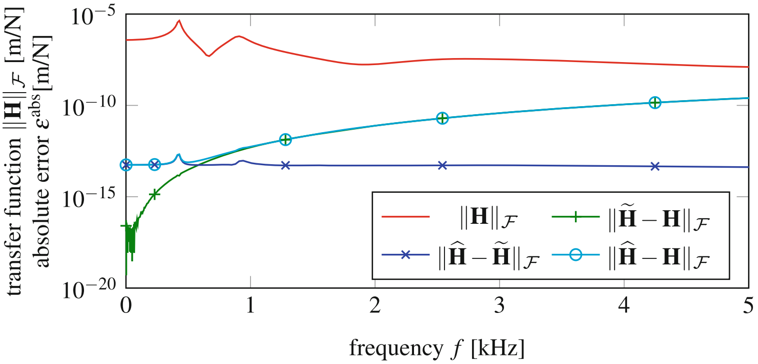


Fig. 30.3 Transfer function of the high order model $\|\mathbf{H}\|_{\mathcal{F}}$, absolute error of the reduced order model with PMOR approach $\|\widetilde{\mathbf{H}} - \mathbf{H}\|_{\mathcal{F}}$, absolute error of the reduced order model with PTMOR approach $\|\widehat{\mathbf{H}} - \mathbf{H}\|_{\mathcal{F}}$, and absolute deviation of PTMOR to PMOR approach $\|\widehat{\mathbf{H}} - \widetilde{\mathbf{H}}\|_{\mathcal{F}}$ using the component mode synthesis approach at $p = 0.5$ and $\varOmega = 13,500$ 1/min

function $\|\mathbf{H}\|_{\mathcal{F}}$ possesses values between 10^{-8} and 10^{-5}. Thus the deviation induced by the additional time dependent matrices is in the range of the reduction error and consequently five to eight magnitudes lower than the frequency response of the high order model. Of course, for faster processes and different parameters the influence of the time dependencies can be much larger.

For identification of the variance in frequency domain depending on the parameter

$$\|\Theta - \Xi\|_{\mathcal{H}_{2,\omega}} = \left(\frac{1}{\pi} \int_{\omega_1}^{\omega_2} \Big(\|\Theta(\imath\omega) - \Xi(\imath\omega)\|_{\mathcal{F}} \Big)^2 \mathrm{d}\omega \right)^{1/2} \tag{30.27}$$

is used, which is related to the frequency-weighted $\mathcal{H}_2$-norm. The integration limits $f_1 = 0$ and $f_2 = 2500\,\text{Hz}$ are used to eliminate the representation of the frequency dependency. This parameter dependent variance is shown in Fig. 30.4, where the seven ticks at the x-axis represent the seven high order models utilized for the interpolation.

The influence of time-dependent terms in the PTMOR approach clearly varies in the parameter space, but is here still in the same magnitude. The magnitude of the influence for the investigated scenario, according to Fig. 30.3, is in an usually negligible range for realistic milling processes.

Should the theoretical influence of the time dependent matrices be demonstrated, the revolution speed can be chosen to very high values. The resulting deviations in frequency domain are obtained with moment matching and shown in Fig. 30.5. Using $\Omega = 0$ yields the PMOR approach. Figure 30.5 depicts a continuously

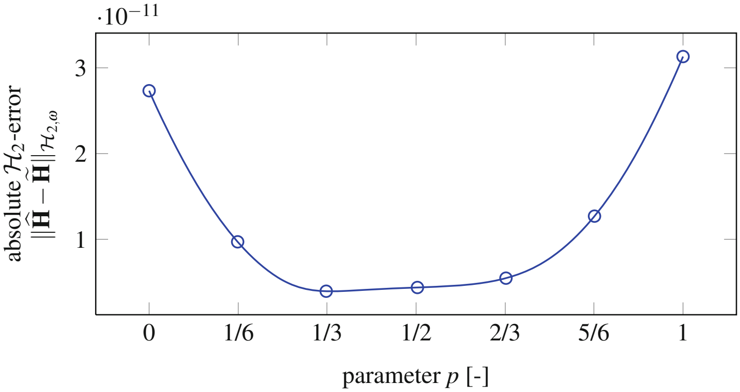


Fig. 30.4 Absolute $\mathcal{H}_2$-error of the PTMOR approach compared to the PMOR approach using the component mode synthesis approach and $\Omega = 13,500$ 1/min

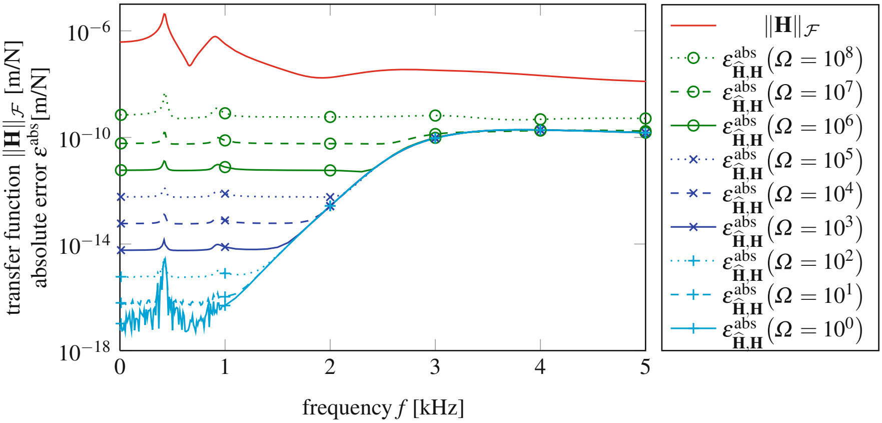


Fig. 30.5 Transfer function of the high order model $\|\mathbf{H}\|_{\mathcal{F}}$ and the absolute deviation of PTMOR approach $\|\widehat{\mathbf{H}} - \mathbf{H}\|_{\mathcal{F}}$ using moment matching at $p = 0.5$ and different revolution speeds Ω

increasing influence of the additional matrices of the PTMOR approach to higher revolution speeds. These additional terms can be taken as correction or maintenance of the repartitioning of kinetic and potential energy, [11].

30.5.2 Time Domain

In the time domain, the milling process is simulated in the presented configuration. Firstly, the milling process is simulated with the PMOR approach and the Matlab integrator dde23 is implemented for constant delays. Secondly, these results are compared with the PTMOR results.

Figure 30.6 shows the time domain results. The upper figures show results of the PMOR approach for each input and output, $x(t)$ and $y(t)$, respectively. The lower figures show the difference of the two approaches. Here only one half rotation of the milling tool is shown due to the almost periodically exciting force. Should a tooth be in cut, higher deviation can be determined, see the characteristic triangular shape of the two deviations. The output $x(t)$ represents the elastic deformation of the T-shaped plate at the milling tool in x direction. It is dominated by the static response of the cutting force, thus the characteristic triangular shape can be seen as well. The deviation of the time dependent terms is clearly visible but here a couple of magnitudes lower than the arising elastic deformations of the workpiece.

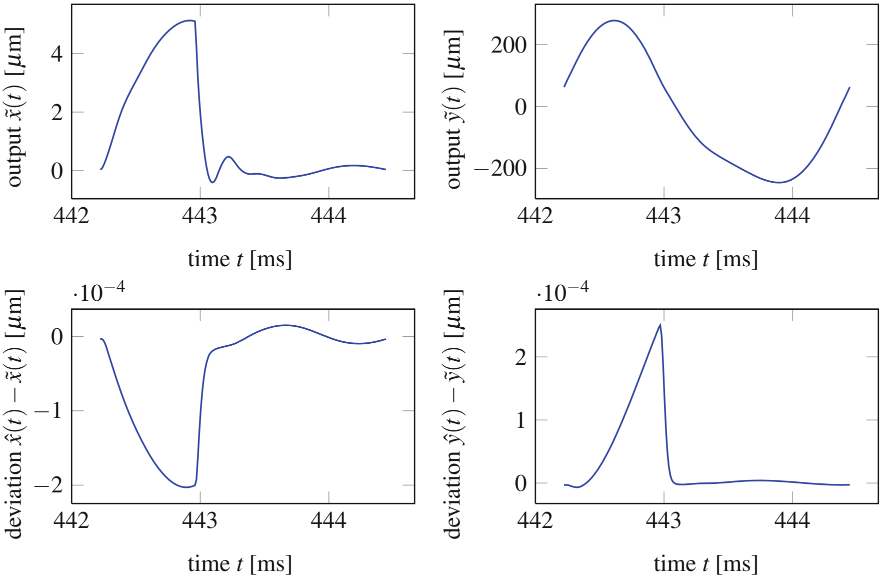


Fig. 30.6 Time domain results of the milling process at $p \approx 1$ and $\Omega = 13,500\,1/\text{min}$ with the reduced order model by component mode synthesis, regarding the elastic deformation of the workpiece in x and y direction as well as the deviation of PTMOR and PMOR approach in x and y direction

30.6 Conclusion

This contribution compares the PTMOR and the PMOR approach for material removal simulations, specifically the milling process of a T-shaped plate. Therefore, different MOR techniques are used and the investigation is conducted in frequency and time domain. Especially the comparison of the two approaches in frequency domain with different revolution speeds provides an impression of the influence of the additional terms of the PTMOR approach.

From the mathematical point of view, the PTMOR approach is the consequential approximation approach of the presented time varying system. However, the approximation error based on neglecting these additional time-dependent terms resulting in the PMOR approach is acceptable for the presented application in the milling process. Machining processes are restricted in cutting speeds due to material properties.

Generally neglecting the additional time-dependent matrices of the PTMOR approach cannot be advised. A case-by-case review is required, since the calculation of the time-dependent matrices of the PTMOR approach can easily be included in the PMOR approach.

References

1. Baumann, M., Eberhard, P.: Interpolation-based parametric model order reduction for material removal in elastic multibody systems. Multibody Sys. Dyn. **39**, 1–16 (2016)
2. Craig, R.: Coupling of substructures for dynamic analyses: an overview. In: Proceedings of the AIAA Dynamics Specialists Conference, Paper-ID 2000-1573, Atlanta, April 5, 2000
3. Fehr, J.: Automated and Error-Controlled Model Reduction in Elastic Multibody Systems. Dissertation, Schriften aus dem Institut für Technische und Numerische Mechanik der Universität Stuttgart, vol. 21. Shaker, Aachen (2011)
4. Fischer, A., Eberhard, P., Ambrósio, J.: Parametric flexible multibody model for material removal during turning. J. Comput. Nonlinear Dyn. **9**(1), 011007 (2014)
5. Hamann, D., Eberhard, P.: Milling stability analysis with varying workpiece dynamics. In: Proceedings of the 4th Joint International Conference on Multibody System Dynamics - IMSD, Montréal (2016)
6. Henninger, C., Eberhard, P., Analysis of dynamic stability for milling processes with varying workpiece dynamics. Proc. Appl. Math. Mech. **8**(1), 10367–10368 (2008)
7. Holzwarth, P., Baumann, M., Volzer, T., Iroz, I., Bestle, P., Fehr, J., Eberhard, P.: Software morembs. University of Stuttgart, Institute of Engineering and Computational Mechanics, Stuttgart (2016)
8. Insperger, T., Stépán, G.: Semi-Discretization for Time-Delay Systems: Stability and Engineering Applications. Springer, New York (2011)
9. Panzer, H., Mohring, J., Eid, R., Lohmann, B.: Parametric model order reduction by matrix interpolation. Automatisierungstechnik **58**, 475–484 (2010)
10. Salimbahrami, B., Lohmann, B.: Order reduction of large scale second-order systems using Krylov subspace methods. Linear Algebra Appl. **415**(2), 385–405 (2006)
11. Tamarozzi, T., Heirman, G.H.K., Desmet, W.: An on-line time dependent parametric model order reduction scheme with focus on dynamic stress recovery. Comput. Methods Appl. Mech. Eng. **268**, 336–358 (2014)
12. Tobias, S.A.: Machine-Tool Vibration. Blackie and Sons, London (1965)
13. Varona, M.C., Geuß, M., Lohmann, B.: p(t)MOR and applications for moving loads. In: Proceedings of the 8th Vienna Conference on Mathematical Modelling (MATHMOD), vol. 48, pp. 677–678 (2015)
14. Zadeh, L.A.: Frequency analysis of variable networks. Proc. IRE **38**(3), 291–299 (1950)