Adaptivity in Model Order Reduction with Proper Orthogonal Decomposition

Dissertation with the aim of achieving a doctoral degree
at the Faculty of Mathematics, Informatics and Natural Sciences
Department of Mathematics
University of Hamburg

submitted by
Carmen Gräßle

Hamburg
2019

Als Dissertation angenommen vom Fachbereich Mathematik der Universität Hamburg auf Grund der Gutachten von:

Prof. Dr. Michael Hinze
Prof. Dr. Peter Benner
Prof. Dr. Stefan Volkwein

Datum der Disputation: 25.06.2019
Vorsitzender der Prüfungskommission: Prof. Dr. Armin Iske

Eidesstattliche Versicherung

Hiermit versichere ich an Eides statt, dass ich die vorliegende Dissertationsschrift selbst verfasst habe und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe.

I, the undersigned, declare upon oath that I have written the dissertation independently and have not used further resources and aids than those stated in the dissertation.

Hamburg, 19. März 2019

Abstract

This thesis is concerned with the approximation of dynamical systems and the optimal control thereof using model order reduction based on *proper orthogonal decomposition (POD)*. The basic idea of this approach is to replace high-fidelity models by low-order surrogates utilizing a Galerkin projection based approach with a non-local basis.

A crucial challenge within *snapshot based* POD model order reduction for time-dependent systems lies in the input dependency. In an *offline phase*, the POD basis is computed from snapshot data obtained by solving the high-fidelity model at several time instances. If a dynamical structure is not captured by the snapshots, this feature will be missing in the reduced-order solution. In this sense, the accuracy of the POD surrogate solution is restricted by how well the snapshots represent the underlying *coherent* structures.

If the process of snapshot sampling is restricted to uniform and static discretizations, this might require very fine resolutions in the whole space-time domain in order to capture important dynamical properties. As a consequence, large-scale systems arise which are computationally expensive or even infeasible to solve.

The objective of this work is to introduce *offline adaptation strategies* which aim to identify problem-specific key dynamics. In the first part, a *snapshot location strategy* for optimal control problems is proposed which utilizes residual based a-posteriori error estimates in order to detect suitable time instances for snapshot generation. In the second part, the inclusion of *spatial adaptivity* in the offline phase for snapshot generation enables to resolve important structures within the spatial domain. We consider the problem setting from an infinite-dimensional view and investigate how POD in Hilbert spaces can be implemented in practice. It is an advantage of this perspective that it only requires the snapshots to live in a common Hilbert space which leads to a great flexibility concerning the actual discretization technique. Moreover, regarding optimal control problems adaptive strategies are crucial in order to adjust the POD model according to the current optimization iterate. In the context of model order reduction for incompressible flows with space-adapted snapshots, two approaches are proposed in order to derive a stable reduced-order model.

Zusammenfassung

Diese Arbeit befasst sich mit der Approximation dynamischer Systeme und der Optimalsteuerung dieser mittels Modellreduktion basierend auf *Proper Orthogonal Decomposition (POD)*. Die grundlegende Idee dieses Ansatzes ist, die hoch-dimensionalen Systeme durch niedrig-dimensionale Modelle mithilfe eines Galerkin Projektions-basierten Ansatzes mit nicht-lokaler Basis zu ersetzen.

Eine wesentliche Herausforderung innerhalb *Snapshot-basierter* POD Modellreduktion für zeitabhängige Systeme liegt in der Abhängigkeit von Eingangsdaten. In einer *offline Phase* wird eine POD Basis bezüglich Snapshots berechnet, welche durch Lösen des hoch-dimensionalen Modells an verschiedenen Zeitpunkten generiert werden. Falls eine dynamische Struktur nicht in den Snapshots erfasst wird, bleibt diese Eigenschaft auch in der reduzierten Lösung fern. In diesem Sinne ist die POD reduzierte Lösung dadurch eingeschränkt, wie gut die Snapshots die zugrunde liegenden *kohärenten* Strukturen erkennen.

Falls der Prozess der Snapshot Generierung auf uniforme und statische Gitter eingeschränkt ist, kann dies zu sehr feinen Auflösungen im gesamten Raum-Zeit Gebiet führen, damit wichtige dynamische Eigenschaften erfasst werden können. Folglich entstehen hoch-dimensionale Systeme, welche rechnerisch aufwändig oder sogar nicht durchführbar sind.

Zielsetzung dieser Arbeit ist, *offline adaptive Strategien* einzuführen, welche das Ziel haben problemspezifische Schlüsseldynamiken zu identifizieren. Im ersten Teil wird eine *Snapshot Platzierungsstrategie* für Optimalsteuerungsprobleme vorgeschlagen, welche residualbasierte a-posteriori Fehlerschätzer verwendet um geeignete Zeitpunkte für die Snapshot Generierung zu bestimmen. Im zweiten Teil ermöglicht die Einbindung von *Ortsadaptivität* in der offline Phase der Snapshot Generierung wichtige Strukturen innerhalb des örtlichen Gebietes aufzulösen. Wir betrachten die Problemstellung aus einer unendlich-dimensionalen Sichtweise und untersuchen wie POD in Hilberträumen praktisch implementiert werden kann. Es ist ein Vorteil dieser Perspektive, dass lediglich erfordert wird, dass die Snapshots in einem gemeinsamen Hilbertraum leben, was eine große Flexibilität bezüglich der tatsächlichen Diskretisierungstechnik zur Folge hat. Außerdem sind im Kontext von Optimalsteuerungsproblemen adaptive Strategien wesentlich um das POD Modell an die aktuelle Optimierungsiterierte anzupassen. Im Zusammenhang der Modellreduktion für inkompressible Strömungen mit ortsadaptiven Snapshots stellen wir zwei Methoden vor um ein stabiles reduziertes Modell herzuleiten.

Publications

Some of the results of this thesis have already been published or submitted.

Section 3 contains results of

- [9] A. Alla, C. Gräßle, and M. Hinze. A residual based snapshot location strategy for POD in distributed optimal control of linear parabolic equations. *IFAC-PapersOnLine*, 49(8):13-18, 2016.
- [10] A. Alla, C. Gräßle, and M. Hinze. Snapshot location for POD in control of a linear heat equation. *PAMM*, 16(1):781-782, 2016.
- [11] A. Alla, C. Gräßle, and M. Hinze. A-posteriori snapshot location for POD in optimal control of linear parabolic equations. *ESAIM:M2AN*, 52(5):1847-1873, 2018.

Section 4 contains results of

- [6] J. O. Alff, C. Gräßle, and M. Hinze. Adaptive trust-region POD for optimal control of the Cahn-Hilliard equation. *PAMM*, 18(1):e201800453, 2018.
- [84] C. Gräßle and M. Hinze. The combination of POD model reduction with adaptive finite element methods in the context of phase field models. *PAMM*, 17(1):47-50, 2017.
- [85] C. Gräßle, and M. Hinze. POD reduced-order modeling for evolution equations utilizing arbitrary finite element discretizations. *ACOM*, 44(6):1941-1978, 2018.
- [86] C. Gräßle, M. Hinze, J. Lang, and S. Ullmann. POD model order reduction with space-adapted snapshots for incompressible flows. *Accepted for publication in ACOM 2019, preprint available https://arxiv.org/abs/1810.03892.*
- [87] C. Gräßle, M. Hinze, and N. Scharmacher. POD for optimal control of the Cahn-Hilliard system using spatially adapted snapshots. In *Numerical Mathematics and Advanced Applications ENUMATH 2017*, 703–711. Springer International Publishing, 2019.

These publications build the fundament for this thesis and are therefore integrated.

Acknowledgement

In the first place, I would like to express my deep gratitude to Prof. Dr. Michael Hinze for his supervision. In particular, I would like to thank for his support, for the mathematical discussions and valuable feedback, for giving interesting research directions and for sharing his expertise and experience. Besides the mathematical context, I would like to thank for the opportunity to participate in many conferences which enabled me to start building a scientific network. Next, I would like to thank Prof. Dr. Peter Benner for his interest in my work, for reviewing my thesis and for evaluating my defense. Moreover, my thanks go to Prof. Dr. Stefan Volkwein for reviewing my thesis. In addition, I would like to thank for encouraging me and awaking my interest in model order reduction and optimal control during my studies at the University Konstanz. Further, I would like to thank Prof. Dr. Armin Iske for including me in his research team. I gratefully acknowledge the financial support of the DFG SPP 1962 Priority Programme *Non-smooth and Complementarity-based Distributed Parameter Systems: Simulation and Hierarchical Optimization*. Next, my thanks go to Dr. Christian Kahle for sharing many codes which I could use as a basis for my own projects and for helping me out with programming questions. Many thanks for carefully proofreading parts of my thesis and the valuable feedback. I would like to thank Prof. Dr. Alessandro Alla for the nice working atmosphere and his great enthusiasm about mathematical research which is contagiously. Moreover, I would like to thank Dr. Sebastian Ullmann for collaborating. I could learn a lot from him especially from his particular precise and detailed working spirit. Many thanks to my colleague Nicolas Scharmacher for carefully proofreading my thesis and his constructive feedback and remarks. Last, my thanks go to my wonderful parents and sister and my amazing boyfriend.

Contents

1 Introduction

1.1 Motivation and challenges

Many computational tasks in science and engineering are numerically expensive, challenging for time-critical applications or problematic in regards to the storage needs.
We think of *multi-query scenarios* in which a model has to be solved many times for different input data or problem settings. An example thereof is *uncertainty quantification (UQ)* where an ensemble of simulations is required to estimate statistical quantities of the solution. Another problem case is *optimization with partial differential equation (PDE) constraints*, where a system of equations has to be solved repeatedly in order to find a minimum of a given cost functional. An application example is shape optimization of an aircraft wing where the aim is to maximize the lift-to-drag ratio, see [78], for instance. The need for good numerical predictions for different material parameters or environmental influences is fundamental, since the financial costs and engineering time for an experimental solution are infeasibly high.
A further class of problems comprises *(near)-real-time scenarios*, where a solution has to be available in a fraction of minutes or seconds. We think of parameter-dependent systems, for which the actual parameter values are not known in advance. An example for such a time-critical application is cancer treatment using thermal ablation, in which the doctor has to adapt the heat delivery in the tissue immediately to the measured patient specific parameter data (see e.g. [175]). A delayed reaction due to computational expenses can lead to serious health effects.
In addition, the issue of *big data*, handling of large-scale information sets as well as data storage and analysis recently has become one of the major research trends. The extraction of dominant structures, autonomous and intelligent learning from data and handling of storage issues require advanced mathematical and numerical concepts.

All of these computational tasks have in common that they need an efficient numerical realization. There are different approaches in order to decrease computational costs.
One option is to *adapt the model fidelity* by e.g. relaxing error tolerances, utilizing coarse mesh resolutions or neglecting model properties. This can be carried out in a hierarchy of an increasingly higher model accuracy when approaching a (local) solution. Examples are multigrid methods (see e.g. [39]) and multilevel trust-region methods (see e.g. [179]). For a general review on adaptive multilevel solution of nonlinear parabolic PDE systems we refer to [125].
Another approach of reducing computational complexity is to make use of *adaptive discretization schemes* which are tailored for the quantity of interest or to the problem's specifications. For example, in the context of simulation and control of multiphase flows utilizing a diffuse interface approach (see e.g. [34, 95, 114]), the use of h-adaptive finite element methods [4, 183] is crucial for a numerical realization due to the steep transitions at the interfacial regions. The inclusion of adaptive discretization techniques usually leads to a notable speed up in comparison to a static discretization while ensuring accuracy. However, computations can still be quite costly.
A different strategy for tackling intensive numerical tasks is to apply *model order reduction techniques*. The basic idea is to replace the high-dimensional problem by a low-order surrogate model with the goal to speed up computations while obtaining solutions with a good approximation quality. The usual procedure of *reduced-order modeling* can be decomposed as shown in Figure 1.

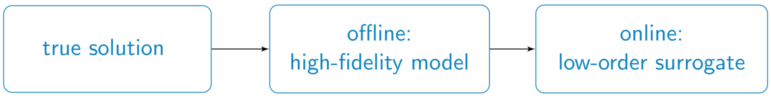

Figure 1: Reduced-order modeling cascade

In an *offline* phase (*training stage*), solution data (so-called *snapshots*) is generated which approximates the true solution. The snapshots can be obtained by e.g. experimental physical measurements or by numerical solutions of the high-dimensional model at different time instances. The latter is computationally intensive, since large-scale problems are solved. From this input data, *reduced spaces* are constructed. Afterwards, in the *online* phase, the surrogate models of low-dimension are utilized in order to compute approximate solutions in a cheap manner. Commonly used model order reduction techniques are (amongst others) *data fitting methods* like kriging [57, 31] or manifold learning [75, 174] which utilize interpolation or regression. Another class of techniques is given by *projection based methods*. Examples are reduced basis methods, rational interpolation [27] or balanced truncation [16]. In this thesis, we focus on model order reduction utilizing *proper orthogonal decomposition (POD)*.

For many problem settings, model order reduction utilizing POD has proven to be a powerful tool in order to reduce time-dependent large-scale systems to surrogate models of low dimension while preserving a good approximation quality. The range of applications of POD model order reduction comprises a broad scope, including linear and nonlinear parabolic equations [120], optimal control of partial differential equations [3, 89, 103, 119, 185] and fluid dynamics [100, 128]. A general introduction to POD reduced-order modeling can be found in [105, 151, 186], for example. The key idea of the POD technique is to apply a Galerkin ansatz, in which the ansatz functions, i.e. the POD modes, contain information about the underlying dynamical system. Following the approach of snapshot based POD in [166, 167], the system information is retrieved from snapshots of the solution trajectory at several time instances, which are generated in a simulation. Large speed up factors can be obtained due to the low dimension of the reduced models. However, a primary challenge is the strong dependence of the reduced model on the input data.

Let us further specify the issue of the *input dependency* of simulation based POD model order reduction. The POD method is utilized in order to construct a basis which is in a certain sense a best-approximation of the snapshot data. In particular, the POD modes are linear combinations of the snapshots with specifically chosen weighting. In this way, the POD basis identifies key dynamics which are captured in the snapshot set. If the snapshots do not catch an important property of the true solution, then the POD model will not be able to reflect this - the POD model will only be able to reflect what is given in the snapshot set. In other words, it is not guaranteed in general that the solution to the reduced-order model is accurate in regards to the true solution. Its accuracy depends on how well the snapshots reflect the true solution. One possible way to address this issue is to consider *online adaptivity concepts*. The basic idea is to utilize an estimator for the accuracy of the reduced-order model. If the reduced model is no longer suitable for the considered dynamics, a reversion to the full-order model is performed and the solution is added to the POD basis or a complete POD basis update is carried out. Some approaches can even avoid a full-order solve. Related references are for example [17, 45, 83, 162]. In this thesis we follow a different approach.

The goal of this work is to address the issue of input dependency of simulation based POD model order reduction by incorporating *offline adaptivity concepts*. The aim is to construct an adaptive, efficient, reliable and autonomous reduced model in the following sense:

- *Adaptive*: The adaptivity is related to problem specific structures. Either adaptive snapshot location strategies are utilized which select suitable time instances for snapshot generation. Or spatially adaptive schemes are used in order to resolve variations in space well.
- *Efficient*: Very fine resolutions in the whole space-time domain are avoided by using adaptive strategies. This leads to a large speed up in the offline phase with only minor accuracy loss.

- *Reliable*: Error estimates are provided in order to track the error of the reduced solution with respect to the true solution. For a prescribed tolerance we can guarantee an according POD reduced-order model accuracy.

- *Autonomous*: We utilize adaptive methods based on error indicators which are related to the problem by residuals, so that no user intervention is needed.

In [28, p. 249ff], adaptive discretization methods and POD reduced-order modeling are considered as two different techniques for reducing computational complexity. In this thesis both approaches are combined in order to exploit the respective advantages. On the one hand, in the offline phase the use of adaptive discretization schemes for snapshot generation can remarkably reduce the offline computation time in comparison to the use of a (very fine) uniform discretization. At the same time a prescribed approximation quality can be ensured. On the other hand, in the online phase, we expect to speed up the computations, since we solve reduced systems of low dimensions.

1.2 Novelty and scope of this work

The achievements of this thesis consist in the incorporation of adaptivity concepts in simulation based model order reduction utilizing POD and address the challenges arising from them. The thesis is split into two major thematic areas: adaptivity with respect to time and adaptivity with respect to space. In particular, the structure of the thesis and the specific contributions are stated as follows:

- In Section 2, we recall basic concepts and introduce notations concerning functional analysis (Section 2.1), adaptive finite element discretization (Section 2.2), optimal control (Section 2.3), model order reduction utilizing POD (Section 2.4) and phase field systems (Section 2.5).

- Section 3 is concerned with adaptivity with respect to time in optimal control. After giving an overview of existing literature concerned with time adaptivity in the context of model order reduction in Section 3.1, we introduce a linear quadratic optimal control problem in Section 3.2 and recall known results concerning existence, uniqueness and regularity of the solution. A reformulation of the optimality system into a biharmonic equation depending on either the state or the adjoint state is presented in Section 3.3 and is based on [82]. In Section 3.4 we give a summary of the a-posteriori error estimation from [82] for the time discretization and specify the space-time finite element discretization for the biharmonic equation in Section 3.5.
  Having elaborated these foundations, we *propose a snapshot location strategy in optimal control for model order reduction utilizing POD* in Section 3.6. The selection of suitable time instances for snapshot generation in Section 3.6.1 is built upon the a-posteriori error estimates from Section 3.4. In an adaptive cycle, a time grid is created by equidistributing the local error contributions of the error indicator for the biharmonic equation on each time interval. Therefore, the resulting adaptive time grid is related to the optimal solution. A coarse spatial resolution is utilized in order to keep computations cheap. This is justified heuristically by a numerical observation in which spatial and temporal discretization decouple for the considered setting. Moreover, at the same time an approximation of the optimal control is produced which can be used as input control for snapshot generation. Error estimates for the error between the true solution and the time-discrete POD solution to the optimal control problem are derived in Section 3.6.2. Numerical examples in Section 3.6.3 emphasize the strength of the snapshot location method in the case of steep transitions in the temporal component.
  In Section 3.7, we focus on the case of optimal control problems with *control constraints*

and located control. We derive an a-posteriori error estimate for the time-discrete solution of the associated biharmonic equation depending on the adjoint variable in the spirit of [82]. Moreover, we provide a snapshot location strategy for POD model reduction, derive error estimates and perform numerical tests. Since a reformulation can only be made with respect to the adjoint state due to the control constraints, we provide a post-processing step based on [120] in order to adapt the time grid such that it is also a suitable time grid for the state.

- Section 4 focuses on adaptivity with respect to space. After presenting a survey of related literature in Section 4.1, we investigate the *combination of POD model order reduction with spatially adapted snapshots* in Section 4.2. The problem setting of an abstract semilinear evolution problem is introduced in Section 4.2.1. Motivated from an infinite-dimensional perspective, we perform the POD method in Section 4.2.2 which utilizes the snapshot gramian. In this approach, the snapshots only need to fulfill the requirement that they live in a common Hilbert space. We provide a numerical strategy how to implement the inner product of finite element snapshots on adapted, non-nested grids. The numerical realization is based on a collision detection, mesh intersection and integration on complex polyhedra using Stoke's formula. The POD reduced-order model is set up in Section 4.3.1, the treatment of the nonlinearity is discussed in Section 4.3.2 and the expression of the solution of the POD reduced system in the full spatial dimension is explained in Section 4.3.3. An error analysis for the error between the true solution and the POD reduced solution is proposed in Section 4.4. Numerical examples for the case of snapshots with nested and non-nested discretizations are presented in Section 4.5.
  We investigate *POD model order reduction with space-adapted snapshots for incompressible flows* in Section 4.6. The problem setting of the Navier-Stokes equations is introduced in Section 4.6.1. We identify the challenge arising in the case of space-adapted snapshots in Section 4.6.2, which consists in the fact that the weak divergence-free property of the snapshots holds only in the respective adapted finite element spaces and is not guaranteed in a common sense. The first solution approach to derive a stable reduced-order model is proposed in Section 4.6.3 and is a velocity reduced-order model in the spirit of [166, 167]. It utilizes an optimal projection onto a weakly divergence-free space, such that the POD basis functions are weakly divergence-free in a common sense. The second approach is proposed in Section 4.6.4 and is a velocity-pressure reduced-order model which utilizes supremizer enrichment functions in the spirit of [22, 160]. The novelty of the work lies in the consideration of space-adapted snapshots and the transfer of the forementioned known concepts to this problem case. The treatment of inhomogeneous Dirichlet and initial data is discussed in Section 4.6.5 and is based on the introduction of specific lifting functions. A numerical study is carried out in Section 4.6.6 and shows that the two approaches lead to stable reduced-order models.
  In Section 4.7 we consider *POD model order reduction with space-adapted snapshots in optimal control for a convective Cahn–Hilliard system*. The problem setting is described in Section 4.7.1. The optimization aim is to control the velocity in the transport term. We show the existence of an optimal control and present a numerical test example in Section 4.7.2.

2 Basic concepts

This chapter recalls basic mathematical concepts which are well-known and relevant for the thesis. It comprises definitions and theorems of functional analysis (Section 2.1), adaptive finite element discretization (Section 2.2), optimal control (Section 2.3), model order reduction utilizing proper orthogonal decomposition (Section 2.4) and phase field systems (Section 2.5). We do not aim to provide a complete presentation of these areas, but focus on specific aspects which are of particular relevance for the scope of this work. References are provided, but the notation might be adapted to the notation within this thesis in order to ensure a consistent presentation.

2.1 Functional analysis

Function spaces

Let us start with the definition of the Lebesgue spaces of integrable functions.

Definition 2.1. ($L^p(\Omega)$ spaces, [101, Definition 1.11])**.** *Let Ω be a Lebesgue measurable set on $\mathbb{R}^d$. We define for $p \in [1, \infty)$ the seminorm*

$$\|u\|_{L^p(\Omega)} := \left(\int_\Omega |u(x)|^p \, dx \right)^{1/p}$$

and

$$\|u\|_{L^\infty(\Omega)} := \operatorname*{ess\,sup}_{x\in\Omega} |u(x)| := \inf\{\alpha \geq 0 : \mu(|u| > \alpha) = 0\}.$$

Now, for $p \in [1, \infty]$ we define the spaces

$$\mathcal{L}^p(\Omega) := \{u : \Omega \to \mathbb{R} \textit{ Lebesgue measurable } : \|u\|_{L^p(\Omega)} < \infty\}.$$

These are not normed spaces since there exist measurable functions $u : \Omega \to \mathbb{R}, u \neq 0$, with $\|u\|_{L^p(\Omega)} = 0$. We use the equivalence relation

$$u \sim v \quad \textit{in } L^p(\Omega) \quad :\Leftrightarrow \quad \|u - v\|_{L^p(\Omega)} = 0 \quad \Leftrightarrow \quad u = v \quad \textit{a.e.}$$

to define $L^p(\Omega) = \mathcal{L}^p(\Omega)/\sim$ as the space of equivalence classes of a.e. identical functions, equipped with the norm $\| \cdot \|_{L^p}$. Finally we define

$$\mathcal{L}^p_{loc}(\Omega) := \{u : \Omega \to \mathbb{R} \textit{ Lebesgue measurable } : u \in \mathcal{L}^p(K) \textit{ for all } K \subset \Omega \textit{ compact}\}$$

and set $L^p_{loc}(\Omega) := \mathcal{L}^p_{loc}(\Omega)/\sim$.

For the theory of partial differential equations, it is helpful to relax the concept of classical differentiability to a weak sense.

Definition 2.2. (Weak partial derivative, [69, Section 5.2.1])**.** *Suppose $u, v \in L^1_{loc}(\Omega)$ and α is a multiindex. We say that v is the α^{th}-weak partial derivative of u, written $D^\alpha u = v$, provided*

$$\int_\Omega u D^\alpha \phi \, dx = (-1)^{|\alpha|} \int_\Omega v \phi \, dx$$

for all test functions $\phi \in C^\infty_c(\Omega)$, i.e. infinitely differentiable functions with compact support in Ω.

The Sobolev space of functions, which have weak derivatives of various orders living in L^p spaces, is defined as follows.

Definition 2.3. ($W^{k,p}(\Omega)$ spaces, [101, Definition 1.14]). *Let $\Omega \subset \mathbb{R}^d$ be open. For $k \in \mathbb{N}_0$, $p \in [1, \infty]$, we define the Sobolev space $W^{k,p}(\Omega)$ by*

$$W^{k,p}(\Omega) = \{u \in L^p(\Omega) : u \textit{ has weak derivatives } D^\alpha u \in L^p(\Omega) \textit{ for all } |\alpha| \leq k\}$$

equipped with the norm

$$\begin{aligned} \|u\|_{W^{k,p}(\Omega)} &:= \left(\sum_{|\alpha| \leq k} \|D^\alpha u\|^p_{L^p(\Omega)} \right)^{1/p}, \quad p \in [1, \infty), \\ \|u\|_{W^{k,\infty}(\Omega)} &:= \sum_{|\alpha| \leq k} \|D^\alpha u\|_{L^\infty(\Omega)}. \end{aligned}$$

For $p = 2$, we write $H^k(\Omega) = W^{k,2}(\Omega)$. We note that $W^{0,p}(\Omega) = L^p(\Omega)$.

Definition 2.4. ($W^{k,p}_0(\Omega)$ spaces, [69, Section 5.2.2.]). *We denote by $W^{k,p}_0(\Omega)$ the closure of $C^\infty_c(\Omega)$ in $W^{k,p}(\Omega)$. We interpret $W^{k,p}_0(\Omega)$ as comprising those functions $u \in W^{k,p}(\Omega)$ such that "$D^\alpha u = 0$ on $\partial\Omega$" for all $|\alpha| \leq k - 1$.*

It is common to write $H^k_0(\Omega) = W^{k,2}_0(\Omega)$.
In the context of time-dependent evolution problems, the concept of abstract function spaces, i.e. function spaces on a bounded interval $[a, b] \subset \mathbb{R}$ with values in a Banach space, constitutes a suitable tool for the description of weak solutions. In particular, we are concerned with the Bochner space of integrable functions.

Definition 2.5. ($L^p(a, b; X)$ spaces, [58, Chapter XVIII, §1.1, Definition 1]). *Let X be a Banach space and $(a, b) \subset \mathbb{R}$ an open set.*
a) We denote by $L^p(a, b; X)$, $p \in [1, \infty)$, the space of (classes of) functions: $t \to u(t) : (a, b) \to X$ such that

i) u is measurable for dt,

ii) $\|u\|_{L^p(a,b;X)} = \left(\int_a^b \|u(t)\|^p_X dt \right)^{1/p} < +\infty.$

b) We denote by $L^\infty(a, b; X)$ the space of (classes of) functions u from (a, b) into X satisfying i) and

ii)' u is bounded almost everywhere over (a,b) and we set $\|u\|_{L^\infty(a,b;X)} = \inf\limits_{\|u(t)\|_X \leq M a.e.} (M).$

Note that $L^2(0, T; \Omega)$ and $L^2(0, T; L^2(\Omega))$ are isometric and isomorphic, i.e.

$$L^2(0, T; \Omega) \cong L^2(0, T; L^2(\Omega)).$$

Next, we consider mappings between function spaces. Let X and Y denote normed real vector spaces and let $L(X, Y)$ denote the space of linear operators $A : X \to Y$ which are bounded in the sense

$$\|A\| := \sup_{x \in X \setminus \{0\}} \frac{\|Ax\|_Y}{\|x\|_X} = \sup_{x \in X, \|x\|_X \leq 1} \|Ax\|_Y < \infty.$$

Theorem 2.6. ([101, Theorem 1.2]). *If Y is a Banach space then $L(X, Y)$ is a Banach space.*

Definition 2.7. (Linear functionals, dual space and dual pairing, [101, Definition 1.5]).

i) Let X be a Banach space. A bounded linear operator $u^* : X \to \mathbb{R}$, i.e. $u^* \in L(X; \mathbb{R})$ is called a bounded linear functional on X.

ii) The space $X^* := L(X, \mathbb{R})$ of linear functionals on X is called dual space of X and is (by Theorem 2.6) a Banach space with the operator norm

$$\|u^*\| := \sup_{x \in X, \|x\|_X \leq 1} |u^*(u)|.$$

iii) We use the notation

$$\langle u^*, u\rangle_{X^*,X} := u^*(u).$$

$\langle \cdot, \cdot\rangle_{X^*,X}$ is called the dual pairing of X^* and X.

We use the notation $H^{-1}(\Omega)$ in order to denote the dual space of $H_0^1(\Omega)$.

The following Riesz representation theorem allows to identify a dual space of a Hilbert space with the Hilbert space itself.

Theorem 2.8. (Riesz representation theorem, [101, Theorem 1.4])**.** *The dual space H^* of a Hilbert space H is isometric to H itself. More precisely, for every $v \in H$ the linear functional u^* defined by*

$$\langle u^*, u\rangle_{H^*,H} := (v, u)_H \quad \forall u \in H$$

is in H^* with norm $\|u^*\|_{H^*} = \|v\|_H$. Vice versa, for any $u^* \in H^*$ there exists a unique $v \in H$ such that

$$\langle u^*, u\rangle_{H^*,H} = (v, u)_H \quad \forall u \in H$$

and $\|u^*\|_{H^*} = \|v\|_H$.

Consequently, the mapping $H^* \to H, u^* \mapsto v$ is an isometric linear isomorphism.

Definition 2.9. (Dual operator, [101, Definition 1.6])**.** *Let X, Y be Banach spaces. Then for an operator $A \in L(X, Y)$ the dual operator $A^* \in L(Y^*, X^*)$ is defined by*

$$\langle A^*u, v\rangle_{X^*,X} = \langle u, Av\rangle_{Y^*,Y} \quad \forall u \in Y^*, v \in X.$$

The space $(X^*)^*$ is called the bidual space or double dual space.

Definition 2.10. (Reflexive space, [101, Definition 1.17])**.** *A Banach space X is called reflexive if the mapping $u \in X \mapsto \langle \cdot, u\rangle_{X^*,X} \in (X^*)^*$ is surjective, i.e. for any $u^{**} \in (X^*)^*$ there exists $u \in X$ with*

$$\langle u^{**}, u^*\rangle_{(X^*)^*,X^*} = \langle u^*, u\rangle_{X^*,X} \quad \forall u^* \in X^*.$$

Remark 2.11. (Reflexive space, [101, Remark 1.8])**.** *L^p is for $1 < p < \infty$ reflexive, since we have the isometric isomorphism $(L^p)^* = L^q$ with $1/p + 1/q = 1$, and thus $((L^p)^*)^* = (L^q)^* = L^p$. Moreover, any Hilbert space is reflexive by the Riesz representation theorem.*

Definition 2.12. (Separable space, [156, II.3])**.** *A metric space which has a countable dense subset is said to be separable.*

Theorem 2.13. (Separable Hilbert space, [156, Theorem II.7])**.** *A Hilbert space H is separable if and only if it has a countable orthonormal basis S. If there are $N < \infty$ elements in S, then H is isomorphic to $\mathbb{C}^N$. If there are countably many elements in S, then H is isomorphic to ℓ_2.*

Definition 2.14. (Gelfand triple, [101, Definition 1.26])**.** *Let H and V be separable Hilbert spaces with continuous dense imbedding $V \hookrightarrow H$. We identify H with its dual H^*. Then, we have the continuous and dense imbeddings*

$$V \hookrightarrow H = H^* \hookrightarrow V^*,$$

which is called Gelfand triple. Note that the imbedding $H \hookrightarrow V^*$ is given by

$$v \in H \mapsto (v, \cdot)_H \in H^* \subset V^*.$$

Often, the Hilbert space H in Definition 2.14 is called pivot space. Other names for the Gelfand triple are evolution triplet or rigged Hilbert space. The inner product $(u, v)_H$ can also be written as $\langle u, v\rangle_{H^*,H} = u(v)$ since we identify $H = H^*$. If $v \in V$, then $u(v) = \langle u, v\rangle_{V^*,V}$ also holds. In conclusion, we have the identity $(u, v)_H = \langle u, v\rangle_{V^*,V}$ for $u \in H$ and $v \in V$. Since V is a Hilbert space one could also identify V with its dual V^*. However, it is not possible to identify H with H^* and V with V^* at the same time, since then one would interpret $u(v)$ also as $(u, v)_V$.

For the description of dynamical systems it makes sense to consider a space which has a Hilbert space structure, is continuous with respect to time t and has finite energy at each time instant.

Definition 2.15. ($W(a, b; V)$ spaces, [58, Chapter XVIII, §1.2, Definition 4, Proposition 6])**.** *Let V be a real, separable Hilbert space. Let $a, b \in \mathbb{R} \cup \{-\infty, +\infty\}$. We denote by $W(a, b; V)$ the space*

$$W(a, b; V) = \{u \in L^2(a, b; V) : u_t \in L^2(a, b; V^*)\}.$$

It is an Hilbert space equipped with the norm

$$\|u\|_{W(a,b;V)} = \left(\|u\|^2_{L^2(a,b;V)} + \|u_t\|^2_{L^2(a,b;V^*)}\right)^{1/2}.$$

Concerning the regularity of elements in the space $W(a, b; V)$, we mention the following property.

Theorem 2.16. ([58, Chapter XVIII, §1.2, Theorem 1])**.** *Let (V, H, V^*) be a Gelfand triple. For $a, b \in \mathbb{R}$, every $u \in W(a, b; V)$ is almost everywhere equal to a continuous function of $[a, b]$ in H. Further, we have*

$$W(a, b; V) \hookrightarrow C^0([a, b]; H),$$

the space $C^0([a, b]; H)$ being equipped with the norm of uniform convergence.

Domains, boundaries and integration formulas

In order to well-define boundary conditions and utilize integration rules according to the Green's formulas (Theorem 2.20), we introduce the Definition 2.17 of a Lipschitz boundary and recall the trace Theorem 2.19.

Definition 2.17. (Lipschitz boundary, [101, Definition 1.13])**.** *Let $\Omega \subset \mathbb{R}^d$ be open and bounded. We say that Ω has a Lipschitz boundary, if for any $x \in \partial\Omega$ there exists $r > 0$, $l \in 1, \ldots, d$, $\sigma \in \{-1, +1\}$ and a function $\gamma \in C^{0,1}(\mathbb{R}^{d-1})$ such that*

$$\Omega \cap B(x; r) = \{y \in B(x; r) : \sigma y_l < \gamma(y_1, \ldots, y_{l-1}, y_{l+1}, \ldots, y_d)\},$$

where $B(x; r)$ denotes the open ball around x with radius r.

Definition 2.18. (Normal derivative, [101, Definition 1.13])**.** *If $\partial\Omega$ is a Lipschitz boundary we define a.e. the unit outer normal field $\vec{n} : \partial\Omega \to \mathbb{R}^d$, where $\vec{n}(x), \|\vec{n}(x)\|_2 = 1$, is the outward pointing unit normal of $\partial\Omega$ at x. We call the directional derivative*

$$\frac{\partial u}{\partial \vec{n}}(x) := \vec{n}(x) \cdot \nabla u(x), \quad x \in \partial\Omega$$

the normal derivative of u.

Theorem 2.19. (Trace theorem, [101, Theorem 1.12])**.** *Assume that $\Omega \subset \mathbb{R}^d$ is open and bounded with Lipschitz boundary. Then for all $p \in [1, \infty]$ there exists a unique bounded linear operator*

$$T : W^{1,p}(\Omega) \to L^p(\partial\Omega)$$

such that

$$Tu = u|_{\partial\Omega} \quad \forall u \in W^{1,p}(\Omega) \cap C(\bar{\Omega}).$$

Here, $\|T\|_{W^{1,p}(\Omega),L^p(\partial\Omega)}$ depends only on Ω and p. Tu is called the trace of u on $\partial\Omega$.

The following theorem is frequently used in the variational theory of partial differential equations.

Theorem 2.20. (Green's formulas, [69, C.1 Theorem 3]). *Assume Ω is a bounded, open subset of $\mathbb{R}^d$ and $\partial\Omega$ is C^1. Let $u, v \in C^2(\bar{\Omega})$. Then*

i) $\displaystyle\int_\Omega \Delta u \, dx = \int_{\partial\Omega} \frac{\partial u}{\partial \vec{n}} \, ds,$

ii) $\displaystyle\int_\Omega Dv \cdot Du \, dx = -\int_\Omega u\Delta v \, dx + \int_{\partial\Omega} \frac{\partial v}{\partial \vec{n}} u \, ds,$

iii) $\displaystyle\int_\Omega u\Delta v - v\Delta u \, dx = \int_{\partial\Omega} u\frac{\partial v}{\partial \vec{n}} - v\frac{\partial u}{\partial \vec{n}} \, ds.$

Lax Milgram and main theorem on monotone operators

The following Lax Milgram theorem delivers existence and uniqueness results for a class of partial differential equations and uses the Riesz representation Theorem 2.8.

Theorem 2.21. (Lax Milgram, [69, §6.2.1, Theorem 1]). *Let H be a real Hilbert space. Assume that $a : H \times H \to \mathbb{R}$ is a bilinear mapping, for which there exist constants $\alpha, \beta > 0$ such that*

i) $|a(u,v)| \leq \alpha \|u\|_H \|v\|_H$ *for $u, v \in H$ (boundedness),*

ii) $\beta \|u\|_H^2 \leq a(u,u)$ *for $u \in H$ (H-coercivity).*

Finally, let $f : H \to \mathbb{R}$ be a bounded linear functional on H. Then there exists a unique element $u \in H$ such that $a(u,v) = \langle f, v\rangle_{H^*,H}$ for all $v \in H$.

Let us proceed by stating theorems which are relevant for the theoretical analysis of some of the considered problems within this thesis. The following definition of a monotone, coercive and hemicontinuous operator is needed in order to formulate the main theorem on monotone operators by Browder and Minty. This is used in Section 3.7 in order to prove existence of a unique weak solution to the considered biharmonic equation.

Definition 2.22. (Monotone operator, [195, Definition 25.2.]). *Let X be a real Banach space, and let $A : X \to X^*$ be an operator. Then A is called monotone iff*

$$\langle Au - Av, u - v\rangle_{X^*,X} \geq 0 \quad \forall u, v \in X.$$

Definition 2.23. (Coercive operator, [195, Definition 25.2.]). *Let X be a real Banach space, and let $A : X \to X^*$ be an operator. Then A is called coercive iff*

$$\lim_{\|u\|_X \to \infty} \frac{\langle Au, u\rangle_{X^*,X}}{\|u\|_X} \to +\infty.$$

Definition 2.24. (Hemicontinuous operator, [195, Definition 26.1]). *Let X be a real Banach space, and let $A : X \to X^*$ be an operator. Then A is said to be hemicontinuous iff the real function*

$$t \mapsto \langle A(u + tv), w\rangle_{X^*,X}$$

is continuous on $[0,1]$ for all $u, v, w \in X$.

Theorem 2.25. (Main theorem on monotone operators, Browder, Minty (1963), [195, Theorem 26.A]). *Let $A : X \to X^*$ be a monotone, coercive and hemicontinuous operator on the real, separable, reflexive Banach space X. Then, for each $b \in X^*$, equation $Au = b$, with $u \in X$ has a solution. The solution set is bounded, convex and closed. If the operator A is strictly monotone, then the equation is uniquely solvable in X.*

Spectral theory of operators

In the following, we provide the definition of a compact operator and the Hilbert-Schmidt theorem, which are needed in the context of POD model order reduction.

Definition 2.26. (Precompact set, [156, VI.5]). *Let (X, d) be a metric space and let $A \subset X$. Then, A is called precompact (totally bounded) if every sequence in A has a subsequence that converges in X.*

Definition 2.27. (Compact operator, [156, VI.5]). *Let X and Y be Banach spaces. An operator $A \in L(X, Y)$ is called compact (or completely continuous) if A takes bounded sets in X into precompact sets in Y. Equivalently, A is compact if and only if for every bounded sequence $\{x_n\} \subset X, \{Ax_n\}$ has a subsequence convergent in Y.*

Example 2.28. (Finite rank operators, [156, VI.5]). *Suppose that the range of A is finite dimensional. That is, every vector in the range of A can be written $Ax = \sum_{j=0}^{n} \alpha_j y^j$, for some fixed family $\{y^j\}_{j=0}^{n}$ in Y. If $\{x_n\}$ is any bounded sequence in X, the corresponding α_j^n are bounded since A is bounded. The usual subsequence trick allows to extract a convergent subsequence from $\{Ax_n\}$ which proves that A is compact.*

Theorem 2.29. (Hilbert-Schmidt theorem, [156, Th. VI.16]). *Let A be a self-adjoint compact operator on the Hilbert space X. Then, there is a complete orthonormal basis $\{\psi_n\}$ for X so that $A\psi_n = \lambda_n \psi_n$ and $\lambda_n \to 0$ as $n \to \infty$.*

Useful inequalities

The following Gronwall's inequality is utilized in order to show energy estimates.

Lemma 2.30. (Gronwall's inequality (differential form), [69, §B.2.j]).

i) Let $\eta(\cdot)$ be a non-negative, absolutely continuous function on $[0, T]$, which satisfies for a.e. t the differential inequality

$$\eta'(t) \leq \phi(t)\eta(t) + \psi(t),$$

where $\phi(t)$ and $\psi(t)$ are non-negative, summable functions on $[0, T]$. Then

$$\eta(t) \leq e^{\int_0^T \phi(s)ds} \left[\eta(0) + \int_0^t \psi(s)ds \right]$$

for all $0 \leq t \leq T$.

ii) In particular, if

$$\eta' \leq \phi\eta \quad \text{on } [0, T] \text{ and } \eta(0) = 0,$$

then $\eta \equiv 0$ on $[0, T]$.

Theorem 2.31. (Young's inequality, [69, §B.2.c]). *Let $1 < p, q < \infty, \frac{1}{p} + \frac{1}{q} = 1$. Then*

$$ab \leq \frac{a^p}{p} + \frac{b^q}{q} \quad (a, b > 0).$$

2.2 Adaptive finite element discretization

The method of finite element discretization has proven to be a powerful tool for the numerical solution of a broad class of partial differential equations. We focus on a continuous Galerkin finite element approach. Starting from a variational formulation of the differential equation in a Sobolev space V, a weak solution is searched in a finite-dimensional trial space $V_h \subset V$, where the test functions live in the same subspace $V_h \subset V$. This is in contrary to Petrov-Galerkin schemes, where different subspaces $V_h, W_h \subset V$ are used as trial and test spaces. A possible construction of such finite-dimensional subspaces V_h can be realized by the method of finite elements. For a comprehensive presentation we refer to [54, 63, 81], for example. The basic idea of the finite element approach is to divide the domain into finite element partitions (*triangulation*) and search for polynomial approximations of the unknown function on each element. Let us introduce an affine equivalent, admissible and regular triangulation in Definition 2.32, which is aligned with the definition given in [4, Section 1.3.3.].

Definition 2.32. (Finite element triangulation)**.** *Let Ω be a polygonal domain with boundary $\partial\Omega$. A finite element partition $\mathcal{T}_h$ of Ω is a collection $\{\mathcal{T}\}$ of elements such that*

(i) The elements form a partition of the domain, that is, $\bar{\Omega} = \bigcup_{\mathcal{T}\in\mathcal{T}_h} \bar{\mathcal{T}}$.

(ii) Each element is a simplex contained in Ω (affine equivalence).

(iii) The non-empty intersection of (the closure) of each distinct pair of elements is either a single common vertex or a single common edge of both elements (admissibility).

(iv) There exists a constant $c \geq c_{\mathcal{T}}$ for all $\mathcal{T} \in \mathcal{T}_h$, where $c_{\mathcal{T}} = h_{\mathcal{T}}/\rho_{\mathcal{T}}$ with $h_{\mathcal{T}}$ being the diameter of the simplex $\mathcal{T}$ and $\rho_{\mathcal{T}}$ is the diameter of the incircle (regularity).

The admissibility condition is often referred to as conformity and excludes the case of so-called *hanging nodes.* The regularity condition means that the smallest angles of all elements is bounded away from zero. Associated with a triangulation $\mathcal{T}_h$ of the domain Ω, we define a finite element space in Definition 2.33.

Definition 2.33. (Finite element space, [4, Section 1.3.4.])**.** *Let p be a non-negative integer and let $\mathcal{T}_h$ be a finite element triangulation as given in Definition 2.32. The finite element subspace of order p associated with the partition $\mathcal{T}_h$ is defined by*

$$V_h = \{v \in C^0(\bar{\Omega}) : \forall \mathcal{T} \in \mathcal{T}_h, v|_{\mathcal{T}} \in \mathbb{P}_p\},$$

where $\mathbb{P}_p$ denotes the space of polynomials up to degree p.

A *Lagrange* or *nodal* basis for the finite-dimensional space V_h is given by the set of functions $\{v_i\}$ with $v_i \in V_h$ and $v_i(P_j) = \delta_{ij}$, where δ_{ij} denotes the Kronecker symbol and $\{P_j\}$ is the set of node points. In case of higher-order finite elements, there are also conditions imposed on the derivatives of the nodal basis. The canonical basis functions have local support, which leads to sparsity in the associated matrix representation of the discrete problem.

We note that an analogous definition of a triangulation can be done utilizing orthotopes instead of simplices. The associated polynomial space which is then used to construct the finite element space is the class $\mathbb{Q}_p$, which denotes the space of polynomials of separate degree p or less. We will utilize such a discretization in Section 3.5.

There are three types of adaptive finite element techniques (see Figure 2):

- *h-adaptivity:* the mesh is refined according to some error indicator (e.g. a-posteriori error estimates, see [4, 37, 183]),
- *p-adaptivity:* the polynomial degree of the nodal basis functions is increased according to the smoothness of the solution (see e.g. [20]),
- *r-adaptivity:* the node points are relocated, but the number of nodes is kept constant. This approach is also known as *moving mesh* approach (see e.g. [107]).

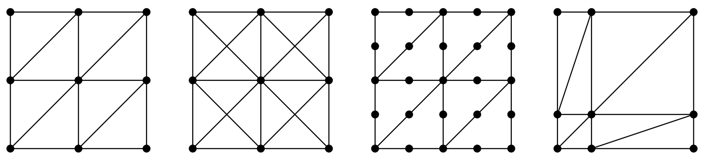

Figure 2: From left to right: Initial mesh, uniform h-refinement, uniform p-refinement, possible r-refinement

We provide details on the h-adaptive finite element approach, since this approach is frequently used in Sections 3 and 4. The usual refinement procedure is visualized in Figure 3 and summarized in a general Algorithm 1. Let us specify each step of the adaptive cycle.

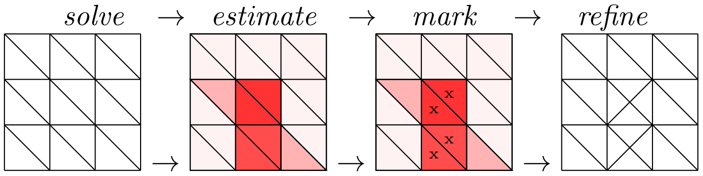

Figure 3: Adaptive finite element cycle

We *initialize* the algorithm with an initial finite element mesh $\mathcal{T}_h^{init}$, which we also refer to as macro-mesh. Each adapted triangulation is a refinement of the macro-mesh. We postulate that no adapted triangulation is coarser than the initial mesh.
Associated with the current triangulation $\mathcal{T}_h$ we construct a finite element space and *solve* the problem using a Galerkin finite element approach.
In order to *estimate* the error between the finite element approximation and the (unknown) solution, we utilize an error indicator based on a residual-type a-posteriori error estimation on each element $\mathcal{T} \in \mathcal{T}_h$.
We determine the triangles which need to be refined by the Dörfler *marking* strategy which is introduced below in Definition 2.34.
As a *refinement* strategy, we use the newest vertex bisection [136], which has the advantage that the resulting grids are nested. The smallest common mesh of two adapted meshes is their overlay [49, 171]. Moreover, the newest vertex bisection conserves the shape regularity of the triangulation.
We note that for time-dependent problems it is often necessary to introduce a coarsening.

Algorithm 1 General h-adaptive finite element algorithm.

```
Input: Initial finite element mesh $\mathcal{T}_h^{init}$, tolerance $\varepsilon > 0$, refinement parameter $\theta \in (0,1)$,
    problem data.
Output: Adapted mesh $\mathcal{T}_h$, discrete problem solution.
Set $\mathcal{T}_h := \mathcal{T}_h^{init}$.
while $\sum_{\mathcal{T} \in \mathcal{T}_h} \eta_{\mathcal{T}} \geq \varepsilon$ do
    Define $V_h$ as the finite element space corresponding to $\mathcal{T}_h$.
    Solve the discrete problem to find a solution in $V_h$.
    Estimate the error contributions using an a-posteriori error estimate $\eta_{\mathcal{T}}$ for each element
      $\mathcal{T} \in \mathcal{T}_h$.
    Mark time intervals according to the Dörfler criterion (Definition 2.34) with marking parameter $\theta$.
    Refine the marked intervals using bisection by newest vertex based on [136].
end while
```

Definition 2.34. (Dörfler marking, [60, strategy (M_*), Section 4.2]). *Mark a set $\mathcal{A} \subset \mathcal{T}_h$ such that*

$$\sum_{\mathcal{T} \in \mathcal{A}} \eta_{\mathcal{T}} \geq (1 - \theta) \sum_{\mathcal{T} \in \mathcal{T}_h} \eta_{\mathcal{T}},$$

for a fixed given value $\theta \in (0, 1)$.

For the computation of error indicators, the jump across an edge plays an important role.

Definition 2.35. (Jumps, [183, Section 1.3.5]). *With every edge E, we associate a unit vector $\vec{n}_E$. For interior edges E its orientation is arbitrary. For any piecewise continuous function ν and any interior edge E, we denote by $\mathbb{J}_E(\nu)$ the jumps of ν across E in the direction of $\vec{n}_E$:*

$$\mathbb{J}_E(\nu)(x) = \lim_{t \to 0_+} \nu(x - t\vec{n}_E) - \lim_{t \to 0_+} \nu(x + t\vec{n}_E) \quad \textit{for all } x \in E.$$

Note that $\mathbb{J}_E(\nu)$ depends on the orientation of $\vec{n}_E$ but that expressions of the form $\mathbb{J}_E(\vec{n}_E \cdot \nabla \nu)$ are independent thereof.

2.3 Optimal control and Lagrange calculus

A comprehensive introduction to the topic of optimization with PDE constraints is given in e.g. [59, 101, 131, 176]. We recall some basic aspects here which are relevant for the scope of this work. Generally spoken, we consider optimal control problems of the abstract form

$$\min_{(y,u) \in Y \times U} J(y, u) \quad \text{s.t.} \quad e(y, u) = 0 \quad \text{and} \quad u \in U_{ad}, \tag{2.1}$$

where Y and U denote the state and control space, respectively, and $J : Y \times U \to \mathbb{R}$ is the *cost functional* (objective functional). By $e : Y \times U \to Z$ we denote an operator related to a partial differential equation (*state equation*) and $U_{ad} \subset U$ is the admissible set of *controls*. Note that various modifications of (2.1) are possible such as the consideration of state constraints or inequality constraints which we do not address further within this scope. We make the following assumptions.

Assumption 2.36. *(i) Z is a Banach space and Y, U are reflexive Banach spaces.*

(ii) $J : Y \times U \to \mathbb{R}$ is continuous, convex and bounded from below.

(iii) $U_{ad} \subset U$ is a closed, convex and non-empty subset.

Supposing that the state equation $e(y, u) = 0$ has a unique solution $y \in Y$ for each control $u \in U_{ad}$, it is common to write y as a function of u, i.e. $y = y(u)$. This enables to derive a reduced problem

$$\min_{u \in U_{ad}} \hat{J}(u) := J(y(u), u), \tag{2.2}$$

where the term *reduced* refers to the elimination of the state variable. We call a control $\bar{u} \in U_{ad}$ a global solution to (2.2) with associated optimal state $\bar{y} = y(\bar{u})$ if

$$\hat{J}(\bar{u}) \leq \hat{J}(u) \quad \text{for all } u \in U_{ad}.$$

Further, $\bar{u} \in U_{ad}$ is called a local solution to (2.2) if there exists a neighborhood $B(\bar{u})$ of $\bar{u}$ in U_{ad} such that

$$\hat{J}(\bar{u}) \leq \hat{J}(u) \quad \text{for all } u \in B(\bar{u}).$$

In order to prove existence of an optimal control, the concepts of weak convergence, weak sequential compactness and lower semicontinuity are important.

Definition 2.37. (Weak convergence, [101, Definition 1.16]). *Let X be a Banach space. We say that a sequence $\{u_k\} \subset X$ converges weakly to $u \in X$, written $u_k \rightharpoonup u$, if*

$$\langle u^*, u_k \rangle_{X^*,X} \to \langle u^*, u \rangle_{X^*,X} \quad \textit{as } k \to \infty \quad \forall u^* \in X^*.$$

Theorem 2.38. (Weak sequential compactness, [101, Theorem 1.17]). *Let X be a reflexive Banach space. Then the following holds*

i) Every bounded sequence $\{x_n\} \subset X$ contains a weakly convergent subsequence, i.e. there are $\{x_{n_i}\} \subset \{x_n\}$ and $x \in X$ with $x_{n_i} \rightharpoonup x$.

ii) Every bounded, closed and convex subset $A \subset X$ is weakly sequentially compact, i.e. every sequence $\{x_n\} \subset A$ contains a weakly convergent subsequence $\{x_{n_i}\} \subset \{x_n\}$ with $x_{n_i} \rightharpoonup x$, where $x \in A$.

Theorem 2.39. (Lower semicontinuity of continuous, convex functionals, [101, Theorem 1.18]). *Let X be a Banach space. Then, any continuous, convex functional $f : X \to \mathbb{R}$ is weakly lower semicontinuous, i.e.*

$$u_k \rightharpoonup u \quad \Rightarrow \quad \liminf_{k\to\infty} f(u_k) \geq f(u).$$

Remark 2.40. *For many optimal control problems, the following steps are carried out in order to show existence of a solution.*

- *Show existence of an infimum $j := \inf\limits_{u \in U_{ad}} \hat{J}(u)$ due to $\hat{J} \geq 0$.*
- *Find a minimizing sequence $\{u_k\} \subset U_{ad}$ with $\hat{J}(u_k) \to j$ for $k \to \infty$.*
- *Find a convergent subsequence $\{u_{k_i}\}$ such that $u_{k_i} \rightharpoonup \bar{u}$ for some $\bar{u} \in U_{ad}$.*
- *Use Theorem 2.39 to show that $\bar{u}$ is an optimal control.*

In order to characterize a local optimal solution, we establish a first-order necessary optimality condition in Definition 2.43. For this, we need the specify the differentiability of the cost functional $J : Y \times U \to \mathbb{R}$ and the operator $e : Y \times U \to Z$.

Definition 2.41. (Directionally, Gâteaux and Fréchet differentiability, [101, Definition 1.29]). *Let $F : U \subset X \to Y$ be an operator with Banach spaces X, Y and $U \neq \emptyset$ open.*

(i) F is called directionally differentiable at $u \in U$ if the limit

$$dF(u, h) = \lim_{t \to 0^+} \frac{F(u + th) - F(u)}{t} \in Y$$

exists for all $h \in X$. In this case, $dF(u, h)$ is called directional derivative of F in the direction h.

(ii) F is called Gâteaux differentiable at $u \in U$ if F is directionally differentiable at u and the directional derivative $F'(u) : X \ni h \mapsto dF(u, h) \in Y$ is bounded and linear, i.e. $F'(u) \in L(X, Y)$.

(iii) F is called Fréchet differentiable at $u \in U$ if F is Gâteaux differentiable at u and if the following approximation condition holds:

$$\|F(u + h) - F(u) - F'(u)h\|_Y = o(\|h\|_X) \quad \textit{for } \|h\|_X \to 0.$$

Assumption 2.42. *(i) $J : Y \times U \to \mathbb{R}$ and $e : Y \times U \to Z$ are continuously Fréchet differentiable.*

(ii) *For all $u \in V$ in a neighborhood $V \subset U$ of U_{ad}, the state equation $e(y,u) = 0$ has a unique solution $y = y(u) \in Y$.*

(iii) *$e_y(y(u),u) \in L(Y,Z)$ has a bounded inverse for all $u \in V \supset U_{ad}$.*

We recall that in (2.2) y is tied to u due to the state equation $e(y,u) = 0$. In particular, using the implicit function theorem (see e.g. [194, Theorem 4B]), the state equation $e(y,u) = 0$ defines locally y as a continuously Fréchet-differentiable function of u (using Assumption 2.42).

Definition 2.43. (First-order necessary optimality condition, [101, Theorem 1.48])**.** *Let Assumptions 2.36 and 2.42 hold. If $\bar{u}$ is a local solution of the reduced problem* (2.2) *then $\bar{u}$ satisfies the variational inequality*

$$\bar{u} \in U_{ad} \quad \textit{and} \quad \langle \hat{J}'(\bar{u}), u - \bar{u}\rangle_{U^*,U} \geq 0 \quad \forall u \in U_{ad}.$$

Note that for a convex problem setting, the first-order necessary optimality condition is sufficient. In order to compute the derivative $\hat{J}'$ of the reduced cost functional, two methods can be followed: the sensitivity approach and the adjoint approach, see e.g. [101, Sections 1.6.1, 1.6.2] or [58, Sections 8, 16]. Let us derive the adjoint gradient representation by a Lagrangian based view. We introduce the Lagrange functional $L : Y \times U \times Z^* \to \mathbb{R}$ by

$$L(y,u,p) = J(y,u) + \langle p, e(y,u)\rangle_{Z^*,Z}.$$

Differentiation leads to

$$\hat{J}'(u) = L_u(y(u),u,p(u)) = J_u(y(u),u) + e_u(y(u),u)^* p(u),$$

where the adjoint state $p = p(u) \in Z^*$ can be computed by solving the adjoint equation given as

$$e_y(y(u),u)^* p = -J_y(y(u),u).$$

For the numerical solution of an optimal control problem, we need to introduce a suitable discretization of the considered variables. There are two different approaches stated in Remark 2.44. Finally, we enumerate possible optimization methods in Remark 2.45.

Remark 2.44. (Discretization concepts)**.**

i) *There are mainly two methodological approaches in order to introduce discrete concepts in PDE constrained optimization: first-discretize-then-optimize and first-optimize-then-discretize. If the same ansatz spaces are used for the state y and the adjoint state p, then both approaches lead to the same discrete optimality system. However, in many situations it makes sense to use a different ansatz space for the state y and adjoint state p, especially if the regularity or the solution properties of the variables differ strongly. A discussion on this issue is given in [101, Section 3.2.4].*

ii) *In [98] the concept of variational discretization is proposed. The idea is not to discretize the space of admissible controls, but to implicitly derive the discretization based on the first-order optimality condition and the discretization of the state space alone.*

Remark 2.45. (Optimization methods)**.** *Depending on the problem setting and the desired convergence rate, there are many different optimization methods available. We like to mention a few:*

- *Descent methods like the projected gradient method (see e.g. [101, Algorithm 2.3] or [59, Algorithm 6]) with projected Armijo rule (see e.g. [101, Section 2.2.2.1] or [59, (5.14)]) or Newton based methods (see e.g. [101, Algorithm 2.5] and [59, Algorithm 3]).*
- *SQP methods (see e.g. [101, Algorithm 2.7] and [59, Algorithm 5]).*
- *Primal-dual active set strategy (see e.g. [59, Algorithm 7]).*

2.4 Model order reduction utilizing proper orthogonal decomposition

The major focus of this thesis lies in the inclusion of adaptivity concepts in the offline phase of model order reduction utilizing proper orthogonal decomposition (POD). The goal of POD based model order reduction is to replace computationally expensive high-fidelity models by low-order surrogates in order to speed up computations and at the same time ensure a good approximation quality. POD model order reduction is a Galerkin projection based method. As trial and test spaces we use reduced spaces which are spanned by POD basis functions. These POD modes have global support and contain information about characteristic properties of the solution. This is in contrary to e.g. finite element methods, where the basis functions are uncorrelated to the physical properties of the underlying dynamical system. The reflection of solution properties within the reduced space is the key feature which enables to use only a small basis length, i.e. reduced spaces of low dimension.
In this section, we recall known concepts and results concerning the POD method (Section 2.4.1) and reduced-order modeling (Section 2.4.2).

2.4.1 Proper orthogonal decomposition

The method of proper orthogonal decomposition (POD) is also known as Karhunen-Loeve expansion, empirical orthogonal functions (EOF) or principal component analysis (PCA) and goes back to [146]. It is utilized in a broad spectrum of disciplines with different applications such as signal analysis and pattern recognition [77], digital picture processing [159, Chapter 5], fluid dynamics and coherent structures [105, 166, 167], data compression [15], meteorology and oceanography [152], control of distributed parameter systems [80] and many more.

Let us assume we are given a set $\mathcal{V} \subset X$. The principle aim within many model order reduction approaches is to approximate the set $\mathcal{V}$ by some n-dimensional linear subspace $X_n \subset X$. A natural question to ask is what is the *best* subspace X_n and *how well* can $\mathcal{V}$ be approximated by X_n. A fundamental measure how well suited a low n-dimensional reduced space is for the approximation of the solution space $\mathcal{V}$ is given by the Kolmogorov n-width according to [117]. Let us consider the deviation of $\mathcal{V}$ from X_n defined by

$$E(\mathcal{V}; X_n) := \sup_{x \in \mathcal{V}} \inf_{y \in X_n} \|x - y\|_X. \tag{2.3}$$

Then, the Kolmogorov n-width of $\mathcal{V}$ in X is defined as

$$d_n(\mathcal{V}, X) := \inf_{X_n \subset X} E(\mathcal{V}; X_n),$$

compare e.g. [150, Definition 1.1]. A small Kolmogorov n-width means that there exists a subspace $X_n \subset X$ which approximates $\mathcal{V}$ well. This is a fundamental prerequisite for a successful model order reduction.

In this thesis, our interest lies in approximating solution trajectories of the form

$$\mathcal{V} = \{y(t) : \; t \in [0, T]\} \subset X. \tag{2.4}$$

As an $L^2(0, T; X)$-analogon to (2.3), we define

$$\mathcal{E}(\mathcal{V}; X_\ell) := \int_0^T \left\| y(t) - \sum_{i=1}^{\ell} (y(t), \psi_i)_X \psi_i \right\|_X^2 dt = \|y(t) - \mathcal{P}^\ell y(t)\|^2_{L^2(0,T;X)}$$

where $\mathcal{P}^\ell$ denotes an orthogonal projection $\mathcal{P}^\ell : X \to X_\ell, \; v \mapsto \sum_{i=1}^{\ell} (v, \psi_i)_X \psi_i$ and $\{\psi_1, \ldots, \psi_\ell\}$ is an orthonormal basis of X_ℓ. The solution to

$$\delta_\ell(\mathcal{V}, X) := \inf_{X_\ell \subset X} \mathcal{E}(\mathcal{V}; X_\ell)$$

is given by the POD space. This means that the POD space creates an optimal ℓ-dimensional subspace with respect to the mean of $\mathcal{V}$, compare [153, Remark 6.3].

Let us now specify how a POD reduced-order space is constructed. The POD method in Hilbert spaces is explained in [89, 121], for example. Here, we recall the main aspects. We specify X as a real separable Hilbert space. The aim of the POD method is to describe the space $\mathcal{V}$ (2.4) by means of few orthonormal functions $\{\psi_i\}_{i=1}^{\ell} \subset X$ with $\ell \leq d := \dim \mathcal{V} \leq \infty$, such that the error between the original trajectory $y(t)$ and its projection onto the ℓ-dimensional subspace $X_\ell = \operatorname{span}\{\psi_1, \dots, \psi_\ell\} \subset X$ is minimized in a mean square sense

$$\min_{\psi_1,\dots,\psi_\ell \in X} \int_0^T \left\| y(t) - \sum_{i=1}^{\ell} (y(t), \psi_i)_X \psi_i \right\|_X^2 dt \quad \text{s.t.} \quad (\psi_i, \psi_j)_X = \delta_{ij} \text{ for } 1 \leq i, j \leq \ell, \tag{2.5}$$

where δ_{ij} denotes the Kronecker symbol.

In practice, the whole trajectory $y(t)$ is not available, but snapshots of the solution at certain instances. These snapshots can be measurement data of physical experiments or finite element samples of the solution trajectory $\mathcal{V}$ on a given time grid, for example. In order to generate information data of the underlying dynamical system, we follow the idea in [166, 167], where numerical solutions at many time instances are computed and stored as *snapshots* of the solution trajectory. From this snapshot data, a suitable orthonormal basis is computed in a least-squares sense which extracts the dominant, *coherent* structures. Assume we are given the snapshots

$$y^0, \dots, y^n \in X. \tag{2.6}$$

We determine a POD basis $\{\psi_1, \dots, \psi_\ell\} \subset X$ of rank ℓ by solving the minimization problem

$$\min_{\psi_1,\dots,\psi_\ell \in X} \sum_{j=0}^{n} \alpha_j \left\| y^j - \sum_{i=1}^{\ell} (y^j, \psi_i)_X \psi_i \right\|_X^2 \quad \text{s.t.} \quad (\psi_i, \psi_j)_X = \delta_{ij} \text{ for } 1 \leq i, j \leq \ell, \tag{2.7}$$

where α_j are non-negative weights. If we consider the case in which $\{y^j\}_{j=0}^n$ are finite element solutions of an evolution equation at time instances $\{t_j\}_{j=0}^n$, then the weights α_j are often chosen as trapezoidal weights, i.e.

$$\alpha_0 = \frac{\Delta t_1}{2}, \quad \alpha_j = \frac{\Delta t_{j+1} + \Delta t_j}{2} \text{ for } j = 1, \dots, n-1, \quad \alpha_n = \frac{\Delta t_n}{2},$$

where we denote by $\Delta t_j = t_j - t_{j-1}$ the length of the time intervals for $j = 1, \dots, n$. A solution to (2.7) is called a rank-ℓ POD basis. For the equality constrained minimization problem (2.7), first-order necessary (and by convexity sufficient) optimality conditions can be derived. For this purpose, we introduce the bounded linear operator $\mathcal{Y} : \mathbb{R}^{n+1} \to X$ by

$$\mathcal{Y}\phi = \sum_{j=0}^{n} \sqrt{\alpha_j} \phi_j y^j \quad \text{for } \phi = (\phi_0, \dots, \phi_n) \in \mathbb{R}^{n+1}. \tag{2.8}$$

Its Hilbert space adjoint $\mathcal{Y}^* : X \to \mathbb{R}^{n+1}$ satisfies $(\mathcal{Y}\phi, \psi)_X = (\phi, \mathcal{Y}^*\psi)_{\mathbb{R}^{n+1}}$ for $\phi \in \mathbb{R}^{n+1}$ and $\psi \in X$ and is given by

$$\mathcal{Y}^*\psi = \begin{pmatrix} (\psi, \sqrt{\alpha_0} y^0)_X \\ \vdots \\ (\psi, \sqrt{\alpha_n} y^n)_X \end{pmatrix} \quad \text{for } \psi \in X. \tag{2.9}$$

We introduce the operator

$$\mathcal{R} := \mathcal{Y}\mathcal{Y}^* : X \to X, \tag{2.10}$$

whose action is given by

$$\mathcal{R}\psi = \sum_{j=0}^{n} \alpha_j (\psi, y^j)_X y^j \quad \text{for } \psi \in X.$$

It can be shown that the operator $\mathcal{R}$ is a linear, bounded, non-negative and self-adjoint operator. Since the image $\mathcal{R}(X) = \operatorname{span}\{y^0, \ldots, y^n\}$ has finite dimension, the operator $\mathcal{R}$ is compact, compare Example 2.28. Therefore the Hilbert-Schmidt Theorem 2.29 can be applied which ensures the existence of a complete orthonormal basis $\{\psi_i\}$ for X and a sequence of corresponding non-negative eigenvalues $\{\lambda_i\}$ with

$$\mathcal{R}\psi_i = \lambda_i \psi_i \quad \text{with } \lambda_1 \geq \ldots \geq \lambda_d > 0 \quad \text{and } \lambda_i = 0 \text{ for all } i > d. \tag{2.11}$$

The following theorem states the necessary (and by convexity sufficient) optimality conditions for problem (2.7) and presents the POD projection error. It is taken from [89, Theorem 2.7], where it is formulated and proved for multiple snapshots.

Theorem 2.46. *Let X be a separable real Hilbert space, $y^0, \ldots, y^n \in X$ and $\mathcal{R} : X \to X$ defined by* (2.10)*. Suppose that $\{\lambda_i\}$ and $\{\psi_i\}$ denote the non-negative eigenvalues and associated orthonormal eigenfunctions of $\mathcal{R}$ satisfying* (2.11)*. Then, for every $\ell \in \{1, \ldots, d\}$ the first ℓ eigenfunctions $\{\psi_i\}_{i=1}^{\ell}$ solve* (2.7)*. Moreover, the value of the cost evaluated at the optimal solution $\{\psi_i\}_{i=1}^{\ell}$ satisfies*

$$\sum_{j=0}^{n} \alpha_j \left\| y^j - \sum_{i=1}^{\ell} (y^j, \psi_i)_X \, \psi_i \right\|_X^2 = \sum_{i=\ell+1}^{d} \lambda_i. \tag{2.12}$$

The basis $\{\psi_i\}_{i=1}^{\ell}$ can alternatively be computed via singular value decomposition (SVD). The SVD of the operator $\mathcal{Y}$ is given by

$$\mathcal{Y} = \sum_{i=1}^{d} \sigma_i \sqrt{\alpha_i} (\cdot, \phi_i)_{\mathbb{R}^{n+1}} \psi_i,$$

where $\sigma_1 \geq \ldots \geq \sigma_d > 0$ is the ordered sequence of singular values of $\mathcal{Y}$ with $\sigma_i = \sqrt{\lambda_i}$ for $i = 1, \ldots, d$. For more details we refer to [156, Theorem VI.17], for instance.

In order to choose the dimension of the reduced space $X_\ell = \operatorname{span}\{\psi_1, \ldots, \psi_\ell\}$, we utilize the following criterion. The information content of a POD basis of rank ℓ relatively to the amount of the information content of all snapshots is given by the ratio of modeled information and total information. It is defined by

$$\Gamma(\ell) := \frac{\sum_{i=1}^{\ell} \lambda_i}{\sum_{i=1}^{d} \lambda_i}.$$

We will choose the POD basis length ℓ such that

$$\ell_{min} = \operatorname{argmin}\{\Gamma(\ell) : \Gamma(\ell) > 1 - p\},$$

for a given value p representing the loss of information.

2.4.2 POD Galerkin reduced-order modeling

After the computation of a POD reduced space $X_\ell \subset X$, we utilize a Galerkin approach in order to derive a reduced-order model. In particular, we start from a variational formulation of the considered differential equation and search for a weak solution in the trial space $X_\ell \subset X$, where the test functions live in the same POD space X_ℓ. This leads to an ℓ-dimensional system of equations for the time-dependent expansion coefficients in the POD Galerkin ansatz for the reduced solution.

POD reduced-order modeling for optimal control

Let us consider the optimal control problem (2.1). Analogously to the two structurally different discretization concepts in Remark 2.44(i), we consider two concepts for reduced-order modeling of optimal control problems given by

first-optimize-then-reduce vs. *first-reduce-then-optimize*

as illustrated in Figure 4. We start with a description of the latter approach, where we first reduce the optimal control problem and then derive the respective optimality system. A discussion of modifications and other approaches is given afterwards.

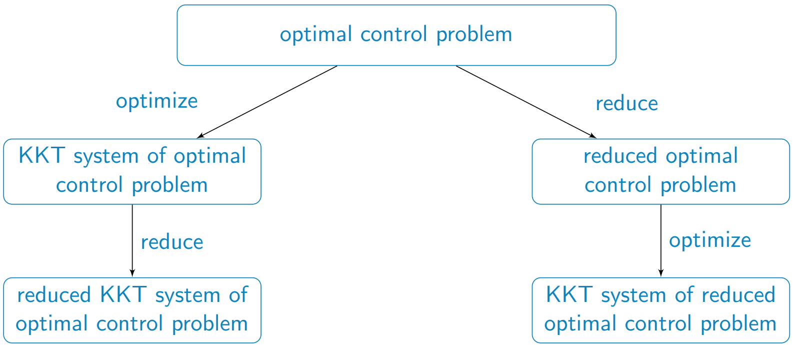
optimal control problem
optimize
reduce
KKT system of optimal control problem
reduced optimal control problem
reduce
optimize
reduced KKT system of optimal control problem
KKT system of reduced optimal control problem

Figure 4: *First-optimize-then-reduce* vs. *first-reduce-then-optimize*

First-reduce-then-optimize. In order to reduce the optimal control problem (2.1), we introduce the reduced state space $Y_\ell \subset Y$. This space can be computed by a POD of a suitable snapshot set (compare Remark 2.47). The reduced optimal control problem reads as

$$\min_{(y_\ell,u_\ell)\in Y_\ell\times U} J_\ell(y_\ell,u_\ell) \quad \text{s.t.} \quad e_\ell(y_\ell,u_\ell)=0 \quad \text{and} \quad u_\ell \in U_{ad} \tag{2.13}$$

with reduced cost functional $J_\ell : Y_\ell \times U \to \mathbb{R}$ and $e_\ell : Y_\ell \times U \to Z$. Note that the control is not reduced yet, but its reduction will be determined by the relation between the reduced adjoint state and control given by the first-order optimality condition. Hence, the reduction of the control is conservative (compare [101, Note 3.2]).
In the weak form of the state equation we search for a POD reduced state $y_\ell \in Y_\ell$ and use the POD space as test space. The reduced adjoint space p_ℓ then will also be in the same POD space by construction, since the POD space is the test space of the state equation. Hence, both reduced state and adjoint variables live in the same POD reduced space. If a solution to (2.1) exists, then by construction there exists a solution to (2.13).

Remark 2.47. (Integrated snapshot set)**.** *If the snapshot set contains pure state information, it is shown in [103, Proposition 4.7 and Remark 4.8(c)] that the error in the adjoint variable depends on a POD projection error of the adjoint. This error can be avoided if one considers a snapshot ensemble of both state and adjoint information. This approach is commonly used, see e.g. [89]. In [116], for example, this is referred to as integrated snapshot set.*

Discussion. It is also imaginable to follow the *first-optimize-then-reduce* approach, where the first-order optimality system is derived and then all infinite-dimensional spaces are replaced by reduced-order spaces. The advantage of this approach is that different reduced spaces can be used for the state and the adjoint state. Especially in the case in which the state and adjoint regularities and dynamics are very different, this approach might better reflect the respective properties. However, one has to be careful in this case: first, one has to prove that the solution of the reduced optimality system is in fact a solution to a respective optimality problem. This question is in general not easy to answer. Moreover, terms of the form

$$\int_\Omega \psi_i^y \, \psi_j^p \, dx$$

will appear, where ψ_i^y denotes a POD basis function corresponding to state snapshots and ψ_j^p is a POD mode computed from adjoint snapshots. This potentially leads to non-symmetric matrices. Moreover, if the POD reduced space for the state and for the adjoint state have different dimensions, the corresponding matrices are rectangular. Thus, solvability has to be ensured. Finally, gradient based optimization methods might not be applicable.

Treatment of nonlinearities

The POD method is a linear procedure, even if the underlying system is nonlinear. This is the strength and weakness of the method. It is well known that the evaluation of nonlinearities in the reduced-order model context is computationally expensive. To make this clear, let us assume we are in a discrete setting and the nonlinear term has the form

$$\Psi^T W f(\Psi\eta(t)),$$

where $\Psi = [\psi_1 \mid \ldots \mid \psi_\ell] \in \mathbb{R}^{N\times\ell}$ is the matrix in which the POD modes are stored columnwise, $W \in \mathbb{R}^{N\times N}$ is a weighting matrix related to the utilized inner product in (2.7) and f describes a nonlinear function. Hence, the treatment of the nonlinearity requires the expansion of the reduced variable $y_\ell(t) = \Psi\eta(t) \in \mathbb{R}^N$ in the full space, then the nonlinearity can be evaluated and finally the result is projected back to the reduced space X_ℓ. Obviously, this means that the reduced model is not fully independent of the high-order dimension N and an efficient simulation cannot be guaranteed. Therefore, it is convenient to seek for *hyper-reduction* methods, i.e. for a treatment of the nonlinearity where the model evaluation cost is related to a low dimension. A possible remedy which is commonly used is given by (discrete) empirical interpolation methods, for which we refer to [25] for EIM, [50] for DEIM and to [61] for Q-DEIM. Another option is given by [12] which investigates nonlinear model reduction via dynamic mode decomposition. Furthermore, in [188] nonlinear model reduction is realized by replacing the nonlinear term by its interpolation into the finite element space. The treatment of the nonlinearity is done in [18] by the missing point estimation method and in [141] by the best points interpolation method.

2.5 Phase field systems

The modeling concept of phase field systems is utilized in order to describe phase separation and the development and evolution of micro structures over time. It is a vivid field of research in many disciplines with a broad spectrum of applications. Examples of such are material sciences, where it is important to know material properties at specific parameter settings and how the process of aging or changes in the problem setting influences them, see e.g. [170]. Another application field appears in image processing, where phase field models are used as tools for the recovery of shapes from images, see e.g. [44].

In the scope of this thesis, a diffuse interface model of the Cahn–Hilliard system is used as an example in which spatial adaptivity is fundamental in order to realize numerical implementations. In this section, we describe the physical phenomenon, introduce the mathematical model and recall known concepts. This enables us to refer to these concepts later.

2.5.1 Physical phenomenon and modeling

We are concerned with the Cahn–Hilliard system which is introduced in [43] as a model for phase separation (*spinodal decomposition*) in binary systems. For a comprehensive study we refer to e.g. [64].

Spinodal decomposition

In physical observations (see e.g. [21]) the following phenomenon has been discovered. Assume we are given a mixture of two partially miscible substances, for example an alloy of two metals (e.g. nickel and iron). At the initial situation, the temperature is greater than the critical temperature. Then, the alloy is quenched (i.e. rapid cooling) to a temperature below the critical temperature. What can be observed is shown in Figure 5 and splits into two stages:

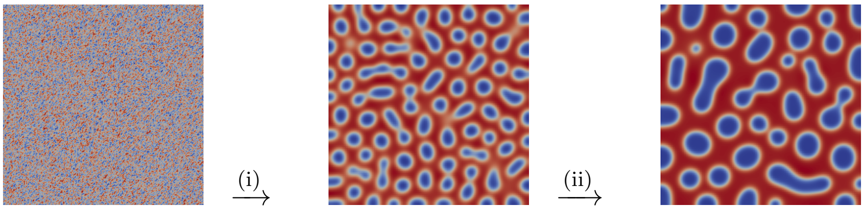

Figure 5: Phase separation (i) and aging (ii)

(i) *Phase separation*: in a rapid demixing phase the mixture decomposes into the pure materials and interfaces appear in between.

(ii) *Aging*: in a slow coarsening phase the length of the interfaces is reduced and patches melt together to larger droplets. Topology changes appear (e.g. coalescence or droplet collisions). An Ostwald-ripening type process can be observed.

Modeling approaches

There are three modeling approaches in order to describe the phase separation process:

- *Atomistic model*: the behavior of single atoms or molecules in the mixture is described. However, a considerable challenge lies in the numerical implementation for realistic scenarios which usually exceeds the scope of usual computing machines due to the large amount of atoms.

- *Sharp interface approach*: the interface is modeled as a lower dimensional manifold. Thus, jumps appear between pure material levels, see Figure 6. It is challenging to work with this modeling approach in case the topology of the phase borders changes.

- *Diffuse interface approach*: a coexistence of molecules from both materials is assumed at the interfacial regions. Thus, the interface is not explicitly defined, but *smeared out*. A *smooth* (needs to be specified mathematically) transition from one to the other material phase is considered, see Figure 6. The advantage of this modeling approach is that topology changes like droplet collision or coalescence can be handled in a natural way. However, in practical simulations the interface width ε is usually larger than the real world thickness.

The relationship between the sharp interface approach and the diffuse interface approach is investigated in the asymptotic analysis for vanishing $\varepsilon \to 0$. For further study on this topic, we refer to e.g. [46, 79, 137]. In this thesis, we concentrate our investigations on the diffuse interface approach.

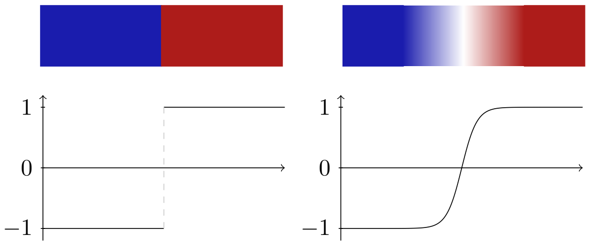

Figure 6: Interface profile: sharp interface approach (left) versus diffuse interface approach (right)

2.5.2 Mathematical model and aspects

Let us consider a binary mixture comprising two substances A and B. In order to describe the spatial distribution over time $(0,T]$ with fixed end time $T>0$, a phase field variable φ is introduced which fulfills $\varphi(t,x)=+1$ in the pure A-phase and $\varphi(t,x)=-1$ in the pure B-phase. Values of φ between -1 and $+1$ represent the interfacial area between the two substances. Its thickness is finite and of order $\mathcal{O}(\varepsilon)$ with $0<\varepsilon\ll 1$. Of course, this setting can be extended to the multi-component case, see e.g. [24, 34, 140]. In the following, we restrict ourselves to the case of binary systems for the sake of clarity.

Ginzburg-Landau energy functional

A phenomenological theory which describes the dynamics of the phase separation is given by the chemical thermodynamics using Gibb's free energy $\mathcal{F}$, see e.g. [147]. Particular mathematical choices for $\mathcal{F}$ are discussed later (see (2.22),(2.23)). The Gibb's free energy of a system describes the difference between the enthalpy and entropy at the current temperature. A system will approach an isothermal equilibrium state that minimizes $\mathcal{F}$. In order to model capillarity (i.e. surface energy of the interface which separates the bulk phases), the gradient term $|\nabla\varphi|$ is additionally taken into account. The resulting free energy E is given by

$$E(\varphi)=\frac{\varepsilon}{2}|\nabla\varphi|^2+\frac{1}{\varepsilon}\mathcal{F}(\varphi). \tag{2.14}$$

The Cahn–Hilliard model in order to describe the equilibrium state of phase separation subject to mass conservation reads as

$$\min\ \mathcal{E}(\varphi)\quad \text{s.t.}\quad \int_\Omega \varphi\, dx = c_{vol}|\Omega|,$$

where $\mathcal{E}(\cdot)$ denotes the Ginzburg-Landau energy functional defined by

$$\mathcal{E}(\varphi)=\int_\Omega E(\varphi)dx=\frac{\varepsilon}{2}\int_\Omega|\nabla\varphi|^2dx+\frac{1}{\varepsilon}\int_\Omega\mathcal{F}(\varphi)dx. \tag{2.15}$$

The first term $\frac{\varepsilon}{2}|\nabla\varphi|^2$ is large for large variations of φ and is minimal if φ is constant. This gives rise to interfacial areas. It can be interpreted as a penalization of jumps and as a regularization term, since it enforces φ to be $H^1(\Omega)$. The second term $\frac{1}{\varepsilon}\mathcal{F}(\varphi)$ is large for $\varphi\notin\{-1,1\}$ and it is minimal for $\varphi\in\{-1,1\}$. Minimization of this term is responsible for phase separation.

Chemical potential

From a chemical point of view, the chemical potential μ describes how the (Gibb's) free energy changes if one of the components of the mixture changes. From a mathematical point of view, the chemical potential can be regarded as first variation of E with respect to the phase field φ, i.e.

$$\mu:=\frac{\delta E}{\delta\varphi}=-\varepsilon\Delta\varphi+\frac{1}{\varepsilon}\mathcal{F}'(\varphi).$$

Cahn–Hilliard equation

There are different approaches to derive the Cahn–Hilliard equation. One option is to consider thermodynamical concepts of mass flux and mass balance law which yields to a diffusion equation, see e.g. [64]. Another derivation can be carried out by computing the H^{-1}-gradient flow of the Ginzburg–Landau free energy, see e.g. [33]. A related model is the Allen–Cahn equation, which can be derived as L^2-gradient flow of the Ginzburg–Landau energy and does not conserve mass [172].

The Cahn–Hilliard equation is a fourth-order parabolic nonlinear partial differential equation. In a bounded and open domain $\Omega \subset \mathbb{R}^d$, $d \in \{1,2,3\}$ with Lipschitz continuous boundary $\partial\Omega$ the Cahn–Hilliard equation reads as

$$\varphi_t + \varepsilon\Delta^2\varphi - \frac{1}{\varepsilon}\Delta\mathcal{F}'(\varphi) = 0 \quad \text{in } (0,T] \times \Omega. \tag{2.16}$$

It is convenient to specify further problem parameters and consider the following model

$$\varphi_t + b\sigma\varepsilon\Delta^2\varphi - \frac{b\sigma}{\varepsilon}\Delta\mathcal{F}'(\varphi) = 0 \quad \text{in } (0,T] \times \Omega. \tag{2.17}$$

By $b > 0$ we denote the mobility which we assume to be constant. It is also possible to consider a degenerate mobility, which is zero in pure phases, see e.g. [23, 66]. The parameter $\sigma > 0$ describes the surface tension and is a material dependent quantity.

Utilizing the definition of the chemical potential, we can decouple the Cahn–Hilliard equation into two coupled partial differential equations of second-order. The coupled Cahn–Hilliard system reads as

$$\varphi_t - b\Delta\mu = 0 \quad \text{in } (0,T] \times \Omega, \tag{2.18a}$$

$$-\sigma\varepsilon\Delta\varphi + \frac{\sigma}{\varepsilon}\mathcal{F}'(\varphi) = \mu \quad \text{in } (0,T] \times \Omega. \tag{2.18b}$$

Remark 2.48. *For numerical implementations, the coupled form* (2.18) *of the Cahn–Hilliard equation is advantageous, since it allows to use finite elements with piecewise continuous, linear ansatz functions and leads to an approximate solution in* $H^1(\Omega)$. *It circumvents the construction of finite element subspaces in* $H^2(\Omega)$ *which would be needed in the numerical consideration of* (2.16). *The construction of such higher-order finite element spaces is more involved, since it requires the basis functions to be continuously differentiable, i.e. in* $C^1(\bar{\Omega})$. *A possible finite element could be the HCT (Hsieh–Clough–Tocher, see e.g. [63, Chapter 8.2]) element, for instance, which has 12 degrees of freedom, or the Agyris element (see e.g. [53, Chapter VII]) with 21 degrees of freedom. For the scope of this thesis, we focus on the numerical treatment of* (2.18).

As boundary conditions we consider homogeneous Neumann conditions

$$\frac{\partial\varphi}{\partial\vec{n}} = 0 \quad \text{and} \quad \frac{\partial\mu}{\partial\vec{n}} = 0 \quad \text{on } [0,T] \times \partial\Omega. \tag{2.19}$$

These boundary conditions are natural and average conserving, i.e. they leave the spatial average of φ invariant in time. Moreover, we impose an initial condition for the phase field

$$\varphi(0,\cdot) = \varphi_0 \quad \text{in } \Omega. \tag{2.20}$$

Bulk free energies

We note that up to now we have assumed that $\mathcal{F}$ is differentiable. However, different choices for the bulk free energy density $\mathcal{F}$ are possible. The free energy function $\mathcal{F}$ shall satisfy the following properties. It is a symmetric function which should be minimal (i.e. $\mathcal{F} \equiv 0$) at $\varphi = \pm 1$, and $\mathcal{F} > 0$ for $\varphi \neq \pm 1$. Moreover, it should hold true that $\lim \mathcal{F}(\varphi) = +\infty$ for $|\varphi| \to \infty$. It turns out that we can split the free energies into a convex ($\mathcal{F}_+$) and a concave ($\mathcal{F}_-$) part as follows

$$\mathcal{F}(\varphi) = \mathcal{F}_+(\varphi) + \mathcal{F}_-(\varphi). \tag{2.21}$$

In particular, we consider three choices for the free energy $\mathcal{F}$, see Figure 7.

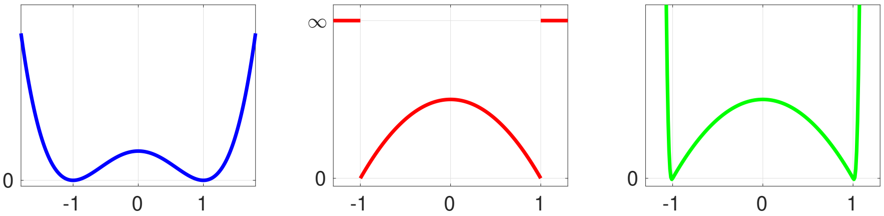

Figure 7: Free energies: double-well $\mathcal{F}_{pol}$ (left), double-obstacle $\mathcal{F}_\infty$ (middle), Moreau–Yosida relaxation $\mathcal{F}_{s,r}$ (right)

The smooth polynomial free energy $\mathcal{F}_{pol}$ (see e.g. [67]) is defined as

$$\mathcal{F}_{pol}(\varphi) = \frac{1}{4}(1-\varphi^2)^2 = \underbrace{\frac{1}{4}\varphi^4}_{\mathcal{F}_+} + \underbrace{(\frac{1}{4} - \frac{1}{2}\varphi^2)}_{\mathcal{F}_-}. \tag{2.22}$$

Another choice is a relaxed double-obstacle free energy

$$\mathcal{F}_{s,r}(\varphi) = \underbrace{\frac{1}{2}(1-\varphi^2)}_{\mathcal{F}_-} + \underbrace{\frac{s}{r}(|\max(0,\varphi-1)|^r + |\min(0,\varphi+1)|^r)}_{\mathcal{F}_+}, \tag{2.23}$$

with relaxation parameter $s \gg 0$ and degree r. The energy $\mathcal{F}_{s,r}$ is a smooth approximation of the non-smooth double-obstacle free energy [35, 143]

$$\mathcal{F}_\infty(\varphi) = \begin{cases} \frac{1}{2}(1-\varphi^2), & \text{if } \varphi \in [-1,1], \\ +\infty, & \text{else,} \end{cases}$$

using a Moreau–Yosida relaxation proposed in [96]. We note that only the double-obstacle free energy $\mathcal{F}_\infty$ enforces $|\varphi| \leq 1$. Both polynomial and relaxed double-obstacle free energies allow φ to have unphysical values $|\varphi| > 1$.

Properties of the Cahn–Hilliard system

We summarize some characteristic properties of the solution φ to the Cahn–Hilliard system.

- *Mass conservation*: the total mass is preserved, i.e.

  $$\frac{d}{dt}\int_\Omega \varphi(x)dx = 0.$$

  By redefining the pure phases we can assume $\int_\Omega \varphi(x)\,dx = 0$ w.l.o.g.

- *Energy dissipation*: it holds true

$$\frac{d}{dt}\int_\Omega \sigma E(\varphi(t))dx = \int_\Omega \left(\frac{\sigma}{\varepsilon}\mathcal{F}'(\varphi(t))\varphi_t(t) + \sigma\varepsilon\nabla\varphi(t)\nabla\varphi_t(t)\right)\, dx = (\mu(t), \varphi_t(t))_{L^2(\Omega)}$$
$$= -\, b(\nabla\mu(t), \nabla\mu(t))_{L^2(\Omega)} = -b\|\nabla\mu(t)\|^2_{L^2(\Omega)} \leq 0,$$

  i.e. the energy decreases in time.

- *Well-posedness*: Existence and uniqueness analysis for different choices for the free energy $\mathcal{F}$ is carried out in e.g. [1, 35, 36, 64, 66, 67]. For example, in the case of the double-well free energy $\mathcal{F}_{pol}$, global existence of a weak solution to (2.18) with $\varphi \in L^\infty(0,T;H^1(\Omega)), \varphi_t \in L^2(0,T;H^1(\Omega)^*), \mu \in L^2(0,T;H^1(\Omega))$ is shown in [64], where also higher regularity results are derived.

Weak form of the Cahn–Hilliard system

Let an initial phase field $\varphi_0 \in H^1(\Omega)$ be given with $\mathcal{E}(\varphi_0) < \infty$. The variational form of the coupled Cahn–Hilliard system (2.18) reads as: find a phase field $\varphi \in W(0,T;H^1(\Omega))$ with $\varphi(0,\cdot) = \varphi_0$ and a chemical potential $\mu \in L^2(0,T;H^1(\Omega))$ such that

$$\frac{d}{dt}(\varphi(t), v)_{L^2(\Omega)} + b(\nabla\mu(t), \nabla v)_{L^2(\Omega)} = 0 \qquad \forall v \in H^1(\Omega), \tag{2.25a}$$
$$\sigma\varepsilon(\nabla\varphi(t), \nabla v)_{L^2(\Omega)} + \frac{\sigma}{\varepsilon}(\mathcal{F}'(\varphi(t)), v)_{L^2(\Omega)} = (\mu(t), v)_{L^2(\Omega)} \qquad \forall v \in H^1(\Omega), \tag{2.25b}$$

holds for almost all $t \in (0,T)$.

Time discretization

In order to find a numerical solution to (2.25), we first introduce a time discretization which allows to use a different (adaptive) finite element space at each time instance for the spatial discretization. As discussed in [70, 71], explicit numerical schemes and most semi-implicit schemes have restrictions on the time steps and force to take time step sizes which are (too) small. Only a few of the fully implicit solvers are gradient stable in the sense that they have a discrete energy that decreases from one time level to the next [108]. A comparison of different time discretization schemes is given in [90].
We consider an unconditional gradient stable time discretization scheme which is based on a convex-concave splitting of the free energy according to [68, 70, 71]. The basic idea is to utilize (2.21) and treat the convex part implicitly and the concave part explicitly, i.e.

$$\mathcal{F}'_+(\varphi^j) + \mathcal{F}'_-(\varphi^{j-1}). \tag{2.26}$$

We introduce a time grid $0 = t_0 < t_1 < \cdots < t_n = T$ with $n \in \mathbb{N}$ and time step sizes $\Delta t_j = t_j - t_{j-1}$ for $j = 1, \ldots, n$. The time-discrete scheme reads as: for given $\varphi^0 = \varphi_0 \in H^1(\Omega)$ find sequences $\varphi^1, \ldots, \varphi^n \in H^1(\Omega)$ and $\mu^1, \ldots, \mu^n \in H^1(\Omega)$ which fulfill the system

$$\left(\frac{\varphi^j - \varphi^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} + b(\nabla\mu^j, \nabla v)_{L^2(\Omega)} = 0 \qquad \forall v \in H^1(\Omega), \tag{2.27a}$$
$$\sigma\varepsilon(\nabla\varphi^j, \nabla v)_{L^2(\Omega)} + \frac{\sigma}{\varepsilon}(\mathcal{F}'_+(\varphi^j) + \mathcal{F}'_-(\varphi^{j-1}), v)_{L^2(\Omega)} = (\mu^j, v)_{L^2(\Omega)} \qquad \forall v \in H^1(\Omega). \tag{2.27b}$$

Space discretization

As mentioned in Remark 2.48, we utilize the coupled form of the Cahn–Hilliard equation in order to use continuous, piecewise linear finite elements for the spatial discretization. In particular, we use mixed finite elements with $\mathbb{P}_1 - \mathbb{P}_1$ nodal basis functions. For more details, we refer to e.g. [65]. We utilize an adaptive discretization scheme following Algorithm 1 and the implementation is based on [114]. Let us denote by V_h^0 the finite element space associated with the initial triangulation $\mathcal{T}_h^{init}$ and φ_h^0 is a suitable approximation of φ_0 in V_h^0. For example, φ_h^0 can be computed as solution to

$$(\varphi_h^0, v)_{L^2(\Omega)} = (\varphi_0, v)_{L^2(\Omega)} \quad \forall v \in V_h^0.$$

The fully discrete Cahn–Hilliard system reads as: for given $\varphi_h^0 \in V_h^0$ find sequences $\varphi_h^1 \in V_h^1, \dots, \varphi_h^n \in V_h^n$ and $\mu_h^1 \in V_h^1, \dots, \mu_h^n \in V_h^n$ which fulfill the system

$$\left(\frac{\varphi_h^j - \varphi_h^{j-1}}{\Delta t}, v \right)_{L^2(\Omega)} + b(\nabla \mu_h^j, \nabla v)_{L^2(\Omega)} = 0 \qquad \forall v \in V_h^j, \tag{2.28a}$$

$$\sigma\varepsilon(\nabla \varphi_h^j, \nabla v)_{L^2(\Omega)} + \frac{\sigma}{\varepsilon}(\mathcal{F}_+'(\varphi_h^j) + \mathcal{F}_-'(\varphi_h^{j-1}), v)_{L^2(\Omega)} = (\mu_h^j, v)_{L^2(\Omega)} \quad \forall v \in V_h^j, \tag{2.28b}$$

where V_h^j denotes the (adapted) finite element space for $j = 1, \dots, n$.

3 Adaptivity with respect to time

Challenge. In order to construct a POD reduced-order model for an optimal control problem which reflects the temporal dynamics of the true optimal solution well, a necessary prerequisite is the representation of the important dynamical features within the snapshot ensemble. A naive solution approach is to use a very fine temporal discretization for snapshot generation in order to identify the key properties in the dynamics. However, this leads to very large computational times, requires storage of data for many time instances and will probably contain redundant information. Moreover, the snapshots are computed with respect to a reference control (e.g. zero control), which might differ strongly from the optimal control. As a consequence, the resulting dynamics of the snapshots might not display a correct temporal evolution with respect to the true solution.

Solution idea. We provide an adaptive snapshot location method for POD reduced-order modeling of linear-quadratic optimal control problems. In order to compute an approximate solution to the infinite-dimensional optimal control problem, we utilize a POD surrogate model which is built from snapshots sampled at specifically chosen time instances. The selection of the snapshot locations is determined by a residual based a-posteriori error estimation for a biharmonic equation which is equivalent to the optimality system. In this way, suitable time instances related to the optimal solution are selected such that the dynamical evolution over time is captured sufficiently well. At the same time an approximation of the optimal control is computed which serves as input control for snapshot generation. In order to achieve fast offline computational times, we utilize a coarse spatial discretization for the solution of the biharmonic equation.

Solution procedure. We reformulate the first-order optimality system of a linear-quadratic optimal control problem as an elliptic equation which is second-order in time and fourth-order in space. This reformulation can be done either with respect to the state or the adjoint state. As a consequence the resulting biharmonic equation depends only on either of these variables, while all other variables are eliminated from the system. For the resulting elliptic equation, standard residual based a-posteriori error estimates can be employed in the spirit of [4] which has been done in [82] for a specific optimal control problem of a linear heat equation. According to the error indicator, an adaptive time grid is constructed which is appropriate with respect to the optimal state or the optimal adjoint state depending on whether the elliptic equation depends on the state or adjoint state.

The novelty of the approach in comparison to existing literature concerned with the reformulation of the optimality system into a biharmonic equation consists in considering a specific control structure (which we refer to as located control) and deriving a-posteriori error estimates in the case of a control constrained optimization problem. In this situation, a reformulation of the optimality system can only be made with respect to the adjoint variable.

In the framework of POD model order reduction, we exploit the construction of the adaptive time grid in the offline phase in order to generate suitable snapshot locations. In particular, this time grid is used to sample the snapshots and to compute the POD reduced-order optimal solution. In order to enable a fast offline computation time, the adaptive time grid is computed with respect to a coarse spatial resolution of the elliptic system. This is possible due to a heuristic observation in our numerical test examples where spatial and temporal discretization decouple for the given problem setting.

Another important aspect of the proposed methodology is that the solution of the elliptic system delivers an approximation of the optimal control with respect to a coarse spatial resolution. This control can be used as input control for snapshot generation.

We provide an error analysis for the proposed methodology which allows to estimate the error between the true optimal solution and the solution to the POD reduced-order control problem.

Outline of the chapter

In Section 3.1 we give an overview of the existing literature concerned with snapshot location strategies in POD reduced-order modeling for dynamical systems. We introduce the linear-quadratic optimal control problem setting in Section 3.2 and reformulate the associated optimality system as an elliptic partial differential equation in Section 3.3. We recall in Sections 3.4 and 3.5 results of [82] regarding the a-posteriori error estimation for the elliptic equation and its discretization using a space-time finite element method. The snapshot location strategy for POD model order reduction in optimal control is presented in Section 3.6. This section includes the description and discussion of the proposed methodology, an error analysis for the state, adjoint state and control variable as well as a numerical example. Section 3.7 focuses on snapshot location in optimal control in the case of located control and considers the inclusion of control constraints. A reformulation into a biharmonic equation is investigated and a residual based a-posteriori error estimation with respect to time is derived for the adjoint state. In Section 3.8 we propose a snapshot location strategy for this case and provide an error analysis for the POD reduced-order solution. The section finishes with a presentation of numerical results. The proposed snapshot location strategy leaves space for further directions of research for which we sketch ideas in Section 3.9 for the consideration of a nonlinear state equation and the application to the model predictive control context.

We note that this section is based on a collaboration with Alessandro Alla and Michael Hinze and some of the results have been published in [9, 10, 11].

3.1 Literature overview

The following publications consider snapshot location in POD model order reduction, presented in a chronological order.
In [118] a medical application of a hyperthermia process for the heating of tumors is considered. Three different sampling strategies of snapshots for POD model order reduction are compared: a uniform time sampling, a logarithmic distribution with *early time sampling* and a reversion thereof leading to a *late time sampling*. Another snapshot sampling scheme is given in [163] in the context of a groundwater flow model. Since the considered model reaches the steady state in an exponential manner, an exponential function is utilized which locates many snapshots early in time and less snapshots later in time.
In [123], POD model reduction for dynamical systems is considered. The choice of time instances for snapshot generation is optimized in order to minimize the error between the solution trajectory and the solution to the POD reduced problem. Starting with an initial uniform snapshot grid, additional snapshots are placed in an optimal way. This leads to a minimization problem subject to nonlinear equality constraints and inequality constraints. Existence of a solution as well as first- and second-order optimality conditions are derived. A continuation of this approach is given in [127], where a complex-valued Helmholtz equation on a frequency band is considered, i.e. a setting of a nonlinear parameter-dependent elliptic system is studied. The optimal snapshot location of [123] is utilized in order to select additional parameters to an initial uniformly distributed parameter set in an optimal way.
In [8], the location of the snapshots is assumed to be fixed. However, a time-adaptive procedure is carried out in order to cluster the snapshots into snapshot sets on sub-intervals according to the ratio of considered singular values versus the sum of all singular values. This leads to the computation of a *time local* POD basis for each sub-interval.
A different approach is proposed in [106] for POD model order reduction of linear and semilinear parabolic equations, where the snapshot locations are selected according to an error equidistribution concept. The global discretization error is estimated by the difference of the solution to an implicit Euler scheme and an implicit trapezoidal rule. For a fixed number of time instances,

the snapshot locations are distributed such that the error indicator is approximately the same in each partition. It is shown that the error equilibration of the full-order model is inherited by the reduced-order model.
An online adaptive snapshot selection strategy called proper snapshot selection (PSS) is proposed in [142]. The most significant snapshots are selected in an incremental procedure by measuring and comparing the energy contained in the respective snapshots using the decay of the eigenvalues as indicator.
In [144], an *on-the-fly* adaptive snapshot selection method is introduced. Within the solution of the full-order model, an incremental POD basis is computed (see e.g. [38, 73]). In particular, the newly selected snapshots are added in order to update the singular value decomposition. The choice of the snapshot locations is controlled by an error criterion which is based on a prediction of the increase in the error of the reduced-order approximation.

To the best of the author's knowledge, the publications [9, 10, 11] are the first contributions concerning snapshot location for POD model order reduction in *optimal control*.

Apart from the computation of suitable time instances for snapshot generation, the method proposed in this chapter also delivers an approximation of the optimal control which can be used as input control for snapshot sampling. Since the dynamics of the state and adjoint state computed from a reference control might differ strongly from the optimal trajectories, a proper choice of the input control is crucial for the accuracy of the reduced-order model. Another option is to update the POD model within the course of the optimization. Concerning this issue, we like to mention the following references. In [3], an *adaptive POD model (APOD)* is proposed, where new snapshots are computed corresponding to the current control iterate within each optimization level. Thus, the POD model is updated according to the dynamics in the current optimization stage.
Within the framework of *trust-region POD* [17, 162], the necessity of a POD basis update is determined by the Carter condition which guarantees relative gradient accuracy.
In [122], the so-called *optimality system POD (OSPOD)* is proposed. The basic idea is to include the equations which determine the POD basis into the optimization process. This makes the optimal control problem more complicated, and thus numerically more intensive. However, it turns out that the computation of a few (projected) gradient steps is sufficient for an initialization.
In [83], we consider two POD basis update strategies: a trust-region POD framework is compared to an OSPOD approach combined with an inexact SQP method with a-posteriori error control according to [177].

3.2 Optimal control problem with distributed control

First, we consider a standard linear-quadratic optimal control problem. This setting is used to explain the reformulation of the associated first-order optimality system into a biharmonic system and the snapshot location strategy in optimal control for POD model order reduction. The inclusion of located control and control constraints is considered in Section 3.7. In addition, the applicability of the proposed method to other problem settings is outlined in Section 3.9.

The state equation

As governing equation (state equation) we consider a linear heat equation given by

$$
\begin{aligned}
y_t - \Delta y &= u \quad \text{in } (0,T] \times \Omega, && (3.1a)\\
y &= 0 \quad \text{on } [0,T] \times \partial\Omega, && (3.1b)\\
y(0,\cdot) &= y_0 \quad \text{in } \Omega. && (3.1c)
\end{aligned}
$$

The function y is in the following called state. We denote by $T > 0$ a fixed end time and $\Omega \subset \mathbb{R}^d, d \in \{1,2,3\}$, is an open bounded domain with Lipschitz boundary $\partial\Omega$ (Definition 2.17). The initial value is denoted by y_0. The source function u will in the following act as the control. It enters the system (3.1) as distributed control on the right-hand side. We assume that we can control within the whole spatial domain. A different kind of control which can only act on parts of the domain (referred to as located control) as well as the inclusion of control constraints are investigated in Section 3.7. As boundary condition we consider homogeneous Dirichlet conditions. Note that we regard a consistent setting in the sense that the initial function complies with the boundary condition. For the analysis and numerical investigation we consider the weak form of (3.1) in the following. For this we make the regularity assumption for the initial condition $y_0 \in L^2(\Omega)$ and for the control function we assume $u \in L^2(0,T;\Omega)$. For a given $u \in L^2(0,T;\Omega)$ we define the functional $\hat{u} : [0,T] \to H^{-1}(\Omega)$ as $v \mapsto \hat{u}(t)v = \int_\Omega u(t,x)v(x)dx$ for $v \in H_0^1(\Omega)$. In this way, we obtain an element $\hat{u} \in L^2(0,T;H^{-1}(\Omega))$.

Definition 3.1 (Weak state solution)**.** *A function $y \in W(0,T;H_0^1(\Omega))$ is called weak solution to* (3.1) *if y satisfies the initial condition* (3.1c) *and the variational equation*

$$\langle y_t(t), v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} + (\nabla y(t), \nabla v)_{L^2(\Omega)} = \langle \hat{u}(t), v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} \tag{3.2}$$

for all $v \in H_0^1(\Omega)$ and almost everywhere in $[0,T]$.

Existence of a unique weak solution $y \in W(0,T;H_0^1(\Omega))$ to the linear heat equation (3.1) is proved in e.g. [69, §7.1.2, Theorems 3 and 4]. Due to

$$W(0,T;H_0^1(\Omega)) \hookrightarrow C([0,T];L^2(\Omega)), \tag{3.3}$$

see Theorem 2.16, we have $y \in C([0,T];L^2(\Omega))$ and therefore equality (3.1c) has a precise meaning. The assignment of boundary values is considered in the sense of traces according to Theorem 2.19.

Applying the differential form of Gronwall's inequality (Lemma 2.30), one can derive the energy estimate

$$\max_{0\le t\le T} \|y(t)\|_{L^2(\Omega)} + \|y\|_{L^2(0,T;H_0^1(\Omega))} + \|y_t\|_{L^2(0,T;H^{-1}(\Omega))} \le c\left(\|\hat{u}\|_{L^2(0,T;H^{-1}(\Omega))} + \|y_0\|_{L^2(\Omega)}\right) \tag{3.4}$$

with a constant $c > 0$ which is independent of $\hat{u}$ and y_0, see e.g. [69, §7.1.2b]. Thus, equation (3.1) is well-posed in the sense of Hadamard (parfaitement bien posé), see [91]. The operator

$$\mathcal{S} : L^2(0,T;H^{-1}(\Omega)) \times L^2(\Omega) \to W(0,T;H_0^1(\Omega)), \quad (\hat{u}, y_0) \mapsto y := \mathcal{S}(\hat{u}, y_0) \tag{3.5}$$

defines a weak solution operator associated with equation (3.1).

Note that due to the continuity of the functional $\hat{u}$, we have

$$\|\hat{u}\|^2_{L^2(0,T;H^{-1}(\Omega))} = \int_0^T \|\hat{u}(t)\|^2_{H^{-1}(\Omega)} dt \le \int_0^T \|u(t)\|^2_{L^2(\Omega)} dt = \|u\|^2_{L^2(0,T;\Omega)}.$$

For the subsequent sections, we need the following higher regularity results, which are taken from [69] and adapted to our notation.

Remark 3.2. (Improved regularity, [69, §7.1.3, Theorem 5])**.** *Assume $y_0 \in H_0^1(\Omega)$ and $u \in L^2(0,T;\Omega)$. Then, the weak solution y to* (3.1) *fulfills*

$$y \in L^2(0,T;H^2(\Omega)) \cap L^\infty(0,T;H_0^1(\Omega)) \text{ with } y_t \in L^2(0,T;\Omega) \tag{3.6}$$

and the energy estimate

$$\operatorname*{ess\,sup}_{0\le t\le T} \|y(t)\|_{H_0^1(\Omega)} + \|y\|_{L^2(0,T;H^2(\Omega))} + \|y_t\|_{L^2(0,T;\Omega)} \le c\left(\|u\|_{L^2(0,T;\Omega)} + \|y_0\|_{H_0^1(\Omega)}\right)$$

holds with a constant $c > 0$ which is independent of u and y_0.

Remark 3.3. (Higher regularity, [69, §7.1.3, Theorem 6]). *Assume* $y_0 \in H^{2m+1}(\Omega)$, $\frac{d^k}{dt^k}u \in L^2(0,T;H^{2m-2k}(\Omega)), k = 0,\dots,m$. *Suppose that the following* m^{th}*-order compatibility conditions hold:*

$$g_0 := y_0 \in H_0^1(\Omega), g_1 := u(0) - \Delta g_0 \in H_0^1(\Omega), \dots, g_m := \frac{d^{m-1}}{dt^{m-1}}u(0) - \Delta g_{m-1} \in H_0^1(\Omega).$$

Then, the weak solution y *to* (3.1) *fulfills*

$$\frac{d^k}{dt^k}y \in L^2(0,T;H^{2m+2-2k}(\Omega)), k = 0,\dots,m+1; \tag{3.7}$$

and we have the estimate

$$\sum_{k=0}^{m+1}\left\|\frac{d^k}{dt^k}y\right\|_{L^2(0,T;H^{2m+2-2k}(\Omega))} \le c\left(\sum_{k=0}^{m}\left\|\frac{d^k}{dt^k}u\right\|_{L^2(0,T;H^{2m-2k}(\Omega))} + \|y_0\|_{H^{2m+1}(\Omega)}\right),$$

the constant $c > 0$ *independent of* u *and* y_0.

The cost functional

The cost functional (objective functional) which shall be minimized is given by

$$J(y,u) := \frac{1}{2}\|y - y_d\|^2_{L^2(0,T;\Omega)} + \frac{\alpha}{2}\|u\|^2_{L^2(0,T;\Omega)}, \tag{3.8}$$

where $y_d \in L^2(0,T;\Omega)$ is a given desired state and $\alpha > 0$ denotes the regularization parameter. Note that the cost functional is non-negative and quadratic. In particular, $(y,u) \mapsto J(y,u)$ is continuous, (strictly) convex and hence weakly lower semicontinuous according to Theorem 2.39. Quadratic cost functionals like (3.8) are known as tracking type costs.

The optimal control problem

The optimal control problem which we consider in the following is formulated as

$$\min_{(y,u)\in W(0,T;H_0^1(\Omega))\times L^2(0,T;\Omega)} J(y,u) \quad \text{s.t.} \quad y = \mathcal{S}(u,y_0). \tag{3.9}$$

This is a linear-quadratic optimal control problem. The optimization goal is to steer the state y, which fulfills the linear heat equation (3.1) in a weak sense, as close as possible to the given desired state y_d and at the same time keep the control costs minimal. The optimal control problem (3.9) has a unique solution, see e.g. [131, §III, 2.1, (2.10)]. The proof follows the usual steps according to Remark 2.40. For the following, we proceed according to the *first-optimize-then-discretize* approach as explained in Remark 2.44(i).

The optimality system

Following the usual Lagrange calculus as recalled in Section 2.3, we derive the first-order necessary (and by convexity sufficient) optimality conditions associated with problem (3.9). They are given by the state equation (3.1) together with the adjoint equation

$$-p_t - \Delta p = y - y_d \quad \text{in } [0,T) \times \Omega, \tag{3.10a}$$
$$p = 0 \quad \text{on } [0,T] \times \partial\Omega, \tag{3.10b}$$
$$p(T,\cdot) = 0 \quad \text{in } \Omega, \tag{3.10c}$$

and the optimality condition (sometimes called gradient equation)

$$\alpha u + p = 0 \quad \text{in } [0,T] \times \Omega. \tag{3.11}$$

Definition 3.4 (Weak adjoint solution). *A function $p \in W(0,T;H_0^1(\Omega))$ is called weak solution to* (3.10) *if p satisfies the final condition* (3.10c) *and the variational equation*

$$-\langle p_t(t), v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} + (\nabla p(t), \nabla v)_{L^2(\Omega)} = \langle \hat{r}(t), v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} \tag{3.12}$$

for all $v \in H_0^1(\Omega)$ and almost everywhere in $[0,T]$, where $\hat{r} : [0,T] \to H^{-1}(\Omega)$ is a functional defined by $v \mapsto \hat{r}(t)v = \int_\Omega (y(t,x) - y_d(t,x))v(x)dx$ for $v \in H_0^1(\Omega)$.

While the state equation (3.1) is an initial boundary value problem and hence a forward in time equation, the adjoint equation (3.10) runs backward in time. Utilizing a simple time transformation with $\tau \in [0,T]$ by $\tilde{p}(\tau,\cdot) := p(T-\tau,\cdot), \tilde{y}(\tau,\cdot) := y(T-\tau,\cdot)$ and $\tilde{y}_d(\tau,\cdot) := y_d(T-\tau,\cdot)$, the following initial boundary value problem can be derived

$$\tilde{p}_t - \Delta\tilde{p} = \tilde{y} - \tilde{y}_d^\tau \quad \text{in } (0,T]\times\Omega, \tag{3.13a}$$

$$\tilde{p} = 0 \quad \text{on } [0,T]\times\partial\Omega, \tag{3.13b}$$

$$\tilde{p}(0,\cdot) = 0 \quad \text{in } \Omega. \tag{3.13c}$$

The existence of a unique weak solution $p \in W(0,T;H_0^1(\Omega))$ to (3.13) as well as stability and regularity results now follow analogue to the results for the state. Back transformation concludes the argumentation. In particular, we have the following estimation

$$\max_{0\le t\le T} \|p(t)\|_{L^2(\Omega)} + \|p\|_{L^2(0,T;H_0^1(\Omega))} + \|p_t\|_{L^2(0,T;H^{-1}(\Omega))} \le c\left(\|u\|_{L^2(0,T;\Omega)} + \|y_0\|_{L^2(\Omega)} + \|y_d\|_{L^2(0,T;\Omega)}\right) \tag{3.14}$$

with a constant $c > 0$ which is independent of u, y_0 and y_d.

Fully discretized optimality system

For the temporal discretization, we use a fully implicit Euler scheme. To this end, we introduce a time grid $0 = t_0 < t_1 < \cdots < t_n = T$ with $n \in \mathbb{N}$ and time step sizes $\Delta t_j = t_j - t_{j-1}$ for $j = 1,\dots,n$. We integrate the differential equations (3.2) and (3.12) from t_{j-1} to t_j for $j = 1,\dots,n$ and approximate the integrals involving the state and adjoint variables by a right-sided rectangle method. For the spatial discretization we introduce a partitioning $\mathcal{T}_h$ of the spatial domain Ω into regular disjoint simplices $\mathcal{T}$ such that $\bar{\Omega} = \cup_{\mathcal{T}\in\mathcal{T}_h}\bar{\mathcal{T}}$ holds true. We use continuous, piecewise linear finite elements, i.e. we introduce the finite element space V_h as

$$V_h = \{v \in C^0(\bar{\Omega}) : v|_{\mathcal{T}} \in \mathbb{P}_1(\mathcal{T}), \mathcal{T} \in \mathcal{T}_h\} \subset H^1(\Omega)$$

and define $V_{h,0} = V_h \cap H_0^1(\Omega)$. Then, the fully discretized state equation reads as follows: find sequences $y_h^0,\dots,y_h^n \in V_h$ such that

$$\left(\frac{y_h^j - y_h^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} + (\nabla y_h^j, \nabla v)_{L^2(\Omega)} = \frac{1}{\Delta t_j}\int_{t_{j-1}}^{t_j} (u(t), v)_{L^2(\Omega)}dt \tag{3.15}$$

and

$$(y_h^0, v)_{L^2(\Omega)} = (y_0, v)_{L^2(\Omega)}$$

holds for all $v \in V_h$. Note that we do not discretize the control, but we make use of variational discretization according to [98], see Remark 2.44(ii). The fully discretized adjoint equation reads as: for given $p_h^n = 0$ find sequences $p_h^{n-1},\dots,p_h^0 \in V_h$ such that

$$\left(\frac{p_h^{j-1} - p_h^j}{\Delta t_j}, v\right)_{L^2(\Omega)} + (\nabla p_h^j, \nabla v)_{L^2(\Omega)} = (y_h^{j-1}, v)_{L^2(\Omega)} - \frac{1}{\Delta t_j}\int_{t_{j-1}}^{t_j} (y_d(t), v)_{L^2(\Omega)}dt \tag{3.16}$$

for all $v \in V_h$. Note that the time integral on the right-hand side can also be approximated by the same rectangle method leading to the approximation

$$\frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} (y_d(t), v)_{L^2(\Omega)} dt \approx (y_d(t_j), v)_{L^2(\Omega)}.$$

Of course, different integration schemes are possible. However, since y_d is a given function, we do not approximate the time integral at this stage in order to avoid the approximation error.

3.3 Reformulation of the optimality system

The optimality system (3.1)-(3.10)-(3.11) can be rewritten into a single elliptic boundary value problem in the space-time domain. This biharmonic equation can be derived with respect to the state variable y and with respect to the adjoint variable p, respectively.

Let us mention some publications which have considered different aspects related to this. In [42] a linear-quadratic optimal control problem is transformed into an elliptic system. Numerical experiments are presented in [138] for distributed and boundary control and the tests consider 1d- and 2d-spatial domains. Subsequent to this work, in [139] control constraints are added to the optimal control problem leading to an optimality system with a pointwise projection formula. The reformulation into a biharmonic system is derived. Due to the projection operator, the elliptic system is non-differentiable. For this reason, the projection formula is regularized by smoothed min/max-functions and the convergence of the associated solution for vanishing regularization parameter is shown. In [193], the optimal control of a Burgers equation is considered. Using the nonlinear Cole–Hopf transformation, the resulting optimality system of linear parabolic equations is transformed into a biharmonic equation. Equivalence of the systems is proved and regularization in case of additional control constraints is applied. In the related publication [115] the Burgers equation is linearized and the associated optimality system is transformed into an elliptic equation in space-time. Of particular importance for this chapter is [82]. In this work, a-priori and a-posteriori error analysis is developed for the time discretization of the state and adjoint biharmonic equation. The a-posteriori error estimate is residual based and gives rise to an adaptive scheme which is related to the optimal state and adjoint state, respectively. This time-adaptive scheme will be the key feature in the following snapshot location selection method for POD in optimal control.
We proceed as follows: first, we state the biharmonic system in strong and weak form for the state and the adjoint state, respectively. Well-posedness of the resulting elliptic systems is discussed. Finally, we note in which sense the biharmonic system is equivalent to the optimality system. The presentation and notation is leaned upon [82].

We introduce the Hilbert space

$$H^{2,1}(0,T;\Omega) = L^2(0,T;H^2(\Omega) \cap H_0^1(\Omega)) \cap H^1(0,T;L^2(\Omega)),$$

equipped with inner product

$$(v,w)_{H^{2,1}(0,T;\Omega)} = \int_0^T \int_\Omega vw + v_t w_t + \nabla v \nabla w + \sum_{i,j=1}^{m} \left(\frac{\partial^2 v}{\partial x_i \partial x_j} \frac{\partial^2 w}{\partial x_i \partial x_j} \right) dxdt$$

and natural norm given by

$$\|v\|_{H^{2,1}(0,T;\Omega)} = \left(\|v\|^2_{L^2(0,T;\Omega)} + \|v_t\|^2_{L^2(0,T;\Omega)} + \|\nabla v\|^2_{L^2(0,T;\Omega)} + \sum_{i,j=1}^{d} \left\| \frac{\partial^2 v}{\partial x_i \partial x_j} \right\|^2_{L^2(0,T;\Omega)} \right)^{1/2}.$$

For the following, we need improved regularity results for the state and adjoint state. If $y_0 \in H_0^1(\Omega)$ and $u \in L^2(0,T;\Omega)$ then we have by Remark 3.2 that the state variable fulfills $y \in H^{2,1}(0,T;\Omega)$. Same applies for the adjoint variable, i.e. it holds $p \in H^{2,1}(0,T;\Omega)$ if $y_d \in L^2(0,T;\Omega)$.

Reformulation of the optimality system with respect to the state

Let us consider the following biharmonic equation in y in strong form

$$\begin{aligned} -y_{tt} + \Delta^2 y + \frac{1}{\alpha} y = \frac{1}{\alpha} y_d &\quad \text{in } (0,T)\times\Omega, && (3.17a)\\ y = 0 &\quad \text{on } [0,T]\times\partial\Omega, && (3.17b)\\ \Delta y = 0 &\quad \text{on } [0,T]\times\partial\Omega, && (3.17c)\\ (y_t - \Delta y)(T,\cdot) = 0 &\quad \text{in } \Omega, && (3.17d)\\ y(0,\cdot) = y_0 &\quad \text{in } \Omega. && (3.17e) \end{aligned}$$

We homogenize (3.17). For this, let the function $\tilde{f} \in H^{2,1}(0,T;\Omega)$ fulfill the boundary conditions (3.17b) and (3.17c) as well as the end time condition (3.17d) and the initial condition (3.17e). We introduce the space

$$H_0^{2,1}(0,T;\Omega) = \{v \in H^{2,1}(0,T;\Omega) : v(0,\cdot) = 0 \text{ in } \Omega\}$$

and set $\tilde{y} := y - \tilde{f}$. Then, $\tilde{y} \in H_0^{2,1}(0,T;\Omega)$ is the solution to the biharmonic equation

$$\begin{aligned} -\tilde{y}_{tt} + \Delta^2 \tilde{y} + \frac{1}{\alpha}\tilde{y} = \frac{1}{\alpha} y_d - \Delta^2 \tilde{f} + \tilde{f}_{tt} - \frac{1}{\alpha}\tilde{f} &\quad \text{in } (0,T)\times\Omega, && (3.18a)\\ \tilde{y} = 0 &\quad \text{on } [0,T]\times\partial\Omega, && (3.18b)\\ \Delta \tilde{y} = 0 &\quad \text{on } [0,T]\times\partial\Omega, && (3.18c)\\ (\tilde{y}_t - \Delta \tilde{y})(T,\cdot) = 0 &\quad \text{in } \Omega, && (3.18d)\\ \tilde{y}(0,\cdot) = 0 &\quad \text{in } \Omega. && (3.18e) \end{aligned}$$

We define a symmetric bilinear form A_T by

$$A_T : H_0^{2,1}(0,T;\Omega) \times H_0^{2,1}(0,T;\Omega) \to \mathbb{R},$$

$$A_T(\tilde{y}, v) = \int_0^T \int_\Omega \left(\tilde{y}_t v_t + \frac{1}{\alpha}\tilde{y} v + \Delta\tilde{y}\Delta v \right) dx dt + \int_\Omega \nabla\tilde{y}(T)\nabla v(T) dx.$$

Furthermore, let us introduce a linear form $\tilde{L}_T$ by

$$\tilde{L}_T : H_0^{2,1}(0,T;\Omega) \to \mathbb{R},$$

$$\tilde{L}_T(v) = \int_0^T \int_\Omega \frac{1}{\alpha} y_d v \, dx dt + \int_0^T \int_\Omega (-\frac{1}{\alpha}\tilde{f} v - \tilde{f}_t v_t - \Delta\tilde{f}\Delta v) dx dt - \int_\Omega \nabla\tilde{f}(T)\nabla v(T) dx.$$

Then, the weak form of the homogenized biharmonic equation (3.18) is given as

$$A_T(\tilde{y}, v) = \tilde{L}_T(v) \quad \forall v \in H_0^{2,1}(0,T;\Omega). \tag{3.19}$$

Existence of a unique solution $\tilde{y} \in H_0^{2,1}(0,T;\Omega)$ to (3.19) follows from the Lax Milgram Theorem 2.21. For this, it is shown in [82, Theorem 2.6] that $A_T : H_0^{2,1}(0,T;\Omega) \times H_0^{2,1}(0,T;\Omega) \to \mathbb{R}$ is a bounded, coercive bilinear mapping. Note that $\tilde{L}_T$ is linear and bounded. Then, $y = \tilde{y} + \tilde{f} \in H^{2,1}(0,T;\Omega)$ is the unique weak solution to (3.17). In particular, $y \in H^{2,1}(0,T;\Omega)$ with $y(0) = y_0 \in H_0^1(\Omega)$ is a weak solution to (3.17), if y satisfies

$$A_T(y, v) = L_T(v) \quad \forall v \in H_0^{2,1}(0,T;\Omega), \tag{3.20}$$

where L_T denotes the linear form defined by

$$L_T : H_0^{2,1}(0,T;\Omega) \to \mathbb{R}, \quad L_T(v) = \int_0^T \int_\Omega \frac{1}{\alpha} y_d v \, dxdt.$$

Note that it is also possible to homogenize (3.1). Then, a reformulation of the associated optimality system leads automatically to a homogenized biharmonic equation.

Reformulation of the optimality system with respect to the adjoint state

Let us consider the following biharmonic equation in p in strong form

$$\begin{aligned} -p_{tt} + \Delta^2 p + \frac{1}{\alpha} p &= -(y_d)_t + \Delta y_d && \text{in } (0,T) \times \Omega, && (3.21a)\\ p &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.21b)\\ \Delta p &= y_d && \text{on } [0,T] \times \partial\Omega, && (3.21c)\\ (p_t + \Delta p)(0,\cdot) &= y_d(0,\cdot) - y_0 && \text{in } \Omega, && (3.21d)\\ p(T,\cdot) &= 0 && \text{in } \Omega. && (3.21e) \end{aligned}$$

We introduce the space $\tilde{H}_0^{2,1}(0,T;\Omega) = \{v \in H^{2,1}(0,T;\Omega) : v(T,\cdot) = 0 \text{ in } \Omega\}$ and define a symmetric bilinear form A_0 by

$$A_0 : \tilde{H}_0^{2,1}(0,T;\Omega) \times \tilde{H}_0^{2,1}(0,T;\Omega) \to \mathbb{R},$$

$$A_0(p,v) = \int_0^T \int_\Omega \left(p_t v_t + \frac{1}{\alpha} p v + \Delta p \Delta v \right) dxdt + \int_\Omega \nabla p(0) \nabla v(0) dx.$$

Furthermore, let us introduce a linear form L_0 by

$$\begin{gathered} L_0 : \tilde{H}_0^{2,1}(0,T;\Omega) \to \mathbb{R}, \\ L_0(v) = \int_0^T \int_\Omega (-(y_d)_t + \Delta y_d) v \, dxdt - \int_\Omega (y_d(0) - y_0) v(0) dx + \int_0^T \int_{\partial\Omega} y_d \nabla v \cdot \vec{n} \, dsdt. \end{gathered} \tag{3.22}$$

The weak formulation for (3.21) is given by

$$A_0(p,v) = L_0(v) \quad \forall v \in \tilde{H}_0^{2,1}(0,T;\Omega). \tag{3.23}$$

If $y_d \in W(0,T;H^1(\Omega))$ the existence of a unique weak solution $p \in \tilde{H}_0^{2,1}(0,T;\Omega)$ to (3.21) follows from the Lax Milgram Theorem 2.21. For this, it is shown in [82, Theorem 2.6] that $A_0 : \tilde{H}_0^{2,1}(0,T;\Omega) \times \tilde{H}_0^{2,1}(0,T;\Omega) \to \mathbb{R}$ is a bounded coercive bilinear mapping and $L_0 : \tilde{H}_0^{2,1}(0,T;\Omega) \to \mathbb{R}$ is linear and bounded.

Equivalence of the optimality system to a biharmonic equation

Theorem 3.5. ([82, Theorem 2.7]). *Let $(y,u) \in W(0,T;H_0^1(\Omega)) \times L^2(0,T;\Omega)$ denote the solution to problem (3.9) with associated adjoint state $p \in W(0,T;H_0^1(\Omega))$. Assume that $y_0 \in H_0^1(\Omega) \cap H^3(\Omega), y_d \in L^2(0,T;\Omega)$ and $u \in L^2(0,T;H^2(\Omega)) \cap H^1(0,T;L^2(\Omega))$. Further, let the compatibility assumption $u(0) + \Delta y_0 \in H_0^1(\Omega)$ hold. Then y satisfies (3.17) a.e. in space-time and is a weak solution to (3.17).*
If $y_d \in H^1(0,T;L^2(\Omega)) \cap L^2(0,T;H^2(\Omega))$ and the compatibility condition $y(T) - y_d(T) + \Delta p(T) \in H_0^1(\Omega)$ is fulfilled, then p solves (3.21) a.e. in space-time and is a weak solution to (3.21).

The proof uses a formal differentiation and insertion of the equations of the optimality system (3.1)-(3.10)-(3.11). This is possible due to the higher regularity assumptions according to Remark 3.3 with $m = 1$. In particular, the state $y \in L^2(0,T;H^4(\Omega)) \cap H^2(0,T;L^2(\Omega)) \cap H^1(0,T;H^2(\Omega))$ fulfills the biharmonic problem (3.17) a.e. in space-time. Then, $y \in H^{2,1}(0,T;\Omega)$ is a weak solution to (3.17). Analogous arguments apply for the adjoint state p. For more details, we refer to [82].

Since equation (3.17) has a unique weak solution and the weak solution y to (3.9) is a weak solution to (3.17) according to Theorem 3.5, the other direction holds, i.e. the weak solution to (3.17) is the optimal state y. The same argumentation holds for the adjoint state p.

We note that the biharmonic equation (3.17) is an equation only with respect to the state variable y with given right-hand side data y_d and y_0. The equation does not explicitly depend on the control variable u or the adjoint variable p. Thus, the optimality system is reduced to an equation depending on a single system variable. The respective observation holds true for the biharmonic equation (3.21) which is an equation with respect to the adjoint variable p and does not explicitly depend on the state y and the control u.

3.4 A-posteriori error estimates for the time discretization

This section gives a summary of the a-posteriori error analysis of [82, Section 3] and a description of the space-time finite element discretization for the biharmonic equations. The residual based a-posteriori error estimation is derived for the time discretization of the biharmonic equation for the state and the adjoint state in the spirit of [4] while keeping the space variable continuous.

Let $0 = t_0 < t_1 < \cdots < t_n = T$ denote a time grid with $n \in \mathbb{N}$ and $\Delta t_j = t_j - t_{j-1}$ and $I_j = [t_{j-1}, t_j]$ for $j = 1, \dots, n$. We define the time-discrete space

$$\mathscr{V}_k = \{v \in H^{2,1}(0,T;\Omega) : v|_{I_j} \in \mathbb{P}_1(I_j)\}, \quad \bar{\mathscr{V}}_k = \mathscr{V}_k \cap H_0^{2,1}(0,T;\Omega).$$

The time-discrete problem consists in finding $y_k \in \mathscr{V}_k$ with $y_k(0,\cdot) = y_0$ and

$$A_T(y_k, v_k) = L_T(v_k) \quad \forall v_k \in \bar{\mathscr{V}}_k. \tag{3.24}$$

There exists a unique solution to the time-discrete problem (3.24) by construction and utilizing Lax–Milgram.

Theorem 3.6. ([82, Theorem 3.3]). *Let $y \in H^{2,1}(0,T;\Omega)$ and $y_k \in \mathscr{V}_k$ denote the solutions to (3.20) and (3.24), respectively. Then, we obtain the a-posteriori error estimation*

$$\|y - y_k\|^2_{H^{2,1}(0,T;\Omega)} \le c\, \eta_y^2, \tag{3.25}$$

with $c > 0$ and

$$\eta_y^2 = \sum_{j=1}^n \Delta t_j^2 \int_{I_j} \left\| \frac{1}{\alpha} y_d + (y_k)_{tt} - \frac{1}{\alpha} y_k - \Delta^2 y_k \right\|^2_{L^2(\Omega)} + \sum_{j=1}^n \int_{I_j} \|\Delta y_k\|^2_{L^2(\partial\Omega)}.$$

A proof of Theorem 3.6 is given in [82] and uses error estimates for Lagrange interpolation, the trace inequality and Young's inequality (Theorem 2.31).
Utilizing the residual based a-posteriori error estimation of Theorem 3.6 leads to a time-adaptive grid which is tailored for the optimal state solution.

Analogously, this can be carried out for the adjoint variable. We define the space

$$\tilde{\mathscr{V}}_k = \mathscr{V}_k \cap \tilde{H}_0^{2,1}(0,T;\Omega).$$

The time-discrete problem consists in finding $p_k \in \tilde{\mathscr{V}}_k$ with $p_k(T,\cdot) = 0$ and

$$A_0(p_k, v_k) = L_0(v_k) \quad \forall v_k \in \tilde{\mathscr{V}}_k. \tag{3.26}$$

Theorem 3.7. ([82, Theorem 3.5]). *Let $p \in \tilde{H}_0^{2,1}(0,T;\Omega)$ and $p_k \in \tilde{\mathscr{V}}_k$ be the solutions to* (3.23) *and* (3.26)*, respectively. Then, we have*

$$\|p - p_k\|^2_{H^{2,1}(0,T;\Omega)} \leq c\, \eta_p^2, \tag{3.27}$$

with $c > 0$ and

$$\eta_p^2 = \sum_{j=1}^{n} \Delta t_j^2 \int_{I_j} \left\| -(y_d)_t + \Delta y_d + (p_k)_{tt} - \frac{1}{\alpha} p_k - \Delta^2 p_k \right\|^2_{L^2(\Omega)} + \sum_{j=1}^{n} \int_{I_j} \|y_d - \Delta p_k\|^2_{L^2(\partial\Omega)}.$$

Utilizing the residual based a-posteriori error estimation of Theorem 3.7 leads to an adaptive time grid which is tailored for the optimal adjoint solution and by the optimality condition (3.11) the adaptive time grid is tailored for the optimal control.

3.5 Space-time mixed finite element discretization

For the numerical solution of the biharmonic equations (3.17) and (3.21), we follow the description in [82, Section 4] and adapt it to our needs. This approach follows the spirit of [55].

We consider a simultaneous space-time mixed finite element discretization scheme. In order to use piecewise linear, continuous finite elements, we decouple the biharmonic equation. We present the discretization for the biharmonic system (3.17) with respect to the state y. The discretization for the biharmonic system (3.21) with respect to the adjoint state p follows along the lines. We introduce an auxiliary variable $w := -\Delta y$. Then, (3.17) can be written as a coupled system in y and w by

$$\begin{aligned}
-y_{tt} - \Delta w + \frac{1}{\alpha} y &= \frac{1}{\alpha} y_d && \text{in } (0,T) \times \Omega, && (3.28a)\\
\Delta y + w &= 0 && \text{in } (0,T) \times \Omega, && (3.28b)\\
y &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.28c)\\
w &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.28d)\\
(y_t - \Delta y)(T,\cdot) &= 0 && \text{in } \Omega, && (3.28e)\\
y(0,\cdot) &= y_0 && \text{in } \Omega. && (3.28f)
\end{aligned}$$

The use of the auxiliary variable allows to use continuous, piecewise linear finite elements and avoids the construction of finite element subspaces in $H^2(\Omega)$ which would be needed for a numerical treatment of the fourth-order in space equation (3.17). Regarding this issue compare Remark 2.48.

We define the function spaces

$$Y := \{v \in H^1(0,T;H_0^1(\Omega)) : v(0,\cdot) = 0 \text{ in } \Omega\}, \; W := L^2(0,T;H_0^1(\Omega)).$$

Then, the mixed variational form for (3.28) reads as: find $y \in H^1(0,T;H_0^1(\Omega))$ with $y(0,\cdot) = y_0$ and $w \in W$ such that

$$\int_0^T \int_\Omega y_t v_t + \frac{1}{\alpha} y v + \nabla w \nabla v \, dx dt + \int_\Omega \nabla y(T) \nabla v(T)\, dx = \int_0^T \int_\Omega \frac{1}{\alpha} y_d v \, dx dt \quad \forall v \in Y, \tag{3.29a}$$

$$\int_0^T \int_\Omega -\nabla y \nabla \phi + w \phi \, dx dt = 0 \qquad \forall \phi \in W. \tag{3.29b}$$

It is shown in [82, Section 4] that the mixed variational system (3.29) admits at most one solution (y, w). Then, it is proved that the pair (y, w) with $w = -\Delta y$ and y denoting the unique solution to (3.17) is a solution to (3.29). As a consequence, the unique solution y to (3.17) coincides with the mixed variational solution to (3.28).

We introduce a quasi-uniform partitioning $\mathscr{T}_h$ of the space-time domain $\Omega_T := (0, T) \times \Omega$ into disjoint regular orthotopes $\mathscr{T}$, such that $\bar{\Omega}_T = \cup_{\mathscr{T} \in \mathscr{T}_h} \bar{\mathscr{T}}$ holds true. We define the finite element space

$$\mathscr{V}_h := \{v \in C^0(\bar{\Omega}_T) : \forall \mathscr{T} \in \mathscr{T}_h, v|_{\mathscr{T}} \in \mathbb{Q}_1\}, \tag{3.30}$$

where $\mathbb{Q}_1$ denotes the space of polynomials of separate degree up to 1. We set $Y_h := \mathscr{V}_h \cap H^1(0, T; H_0^1(\Omega))$, $Y_{h,0} := \mathscr{V}_h \cap Y$ and $W_{h,0} = \mathscr{V}_h \cap W$. Then the fully discrete approximation to (3.29) reads as: find $y_h \in Y_h$, $w_h \in W_{h,0}$ with $y_h(0) = y_0$ such that

$$\int_0^T \int_\Omega (y_h)_t v_t + \frac{1}{\alpha} y_h \, v + \nabla w_h \nabla v dx dt + \int_\Omega \nabla y_h(T) \nabla v(T) dx = \int_0^T \int_\Omega \frac{1}{\alpha} y_d \, v \, dx dt \quad \forall v \in Y_{h,0}, \tag{3.31a}$$

$$\int_0^T \int_\Omega -\nabla y_h \nabla \phi + w_h \phi \, dx dt = 0 \qquad \forall \phi \in W_{h,0}. \tag{3.31b}$$

Now, we propose an adaptive finite element method tailored for the state variable which utilizes the residual based a-posteriori error estimate (3.25) in order to adapt the time discretization to the underlying temporal dynamics. We note that the refinement strategy is based on the error indicator (3.25) which is discrete in time but continuous in space. In this way, we tailor the discretization with respect to the time variable and neglect errors stemming from the spatial variable. This is heuristically justified by the observation in the numerical tests (Section 3.6.3) that for the considered problem setting the temporal discretization is insensitive to the spatial discretization. For details on a-priori error estimations for the fully discretized space-time variational formulation, we refer to [82, Theorem 4.1 and Remark 4.2].

The adaptive finite element method is described in Algorithm 2. We follow the usual adaptivity cycle

$$\textit{solve} \to \textit{estimate} \to \textit{mark} \to \textit{refine}, \tag{3.32}$$

compare Section 2.2. Since we aim to perform an adaptive cycle with respect to the time variable, we choose a structured initial space-time mesh $\mathscr{T}_h^{init}$ which is a *Cartesian product* of the spatial and temporal domain. In particular, we choose an initial time grid $0 = t_0 < \cdots < t_n = T$ with $n + 1$ time points and discretize the spatial domain into d-orthotopes, i.e. intervals for $d = 1$, rectangles for $d = 2$ and cuboids for $d = 3$, respectively. The advantage of this choice is that hanging nodes like shown in Figure 8 (middle) are avoided and the bisection of a temporal interval leads to a structured space-time finite element mesh by construction. Moreover, in such a space-time discretization each space-time block corresponds to a time step of a semi-discretized problem and meshes like shown in Figure 8 (right) are avoided. In particular, the mesh is partitioned in space-time slabs $(t_{j-1}, t_j) \times \Omega$.

As finite element basis functions, we use $\mathbb{Q}_1$ nodal functions, i.e. polynomials in space-time with separate degree up to 1 and local support on the space-time elements. We obtain the adapted space-time grids by refining the initial space-time grid with respect to time, while the discretization of the spatial domain stays the same.

Let us specify the adaptive cycle (3.32) in Algorithm 2. In each level of the adaptive cycle, the variational system (3.31) is *solved*, for which we use a direct solver. The error *estimation* relies on the error indicator (3.25). In particular, for each time interval $I_j, j = 1, \ldots, n$ we add all residual contributions of each spatial domain and neglect the spatial discretization error.

In contrary to [82] where a bulk marking strategy is followed, we utilize the Dörfler criterion

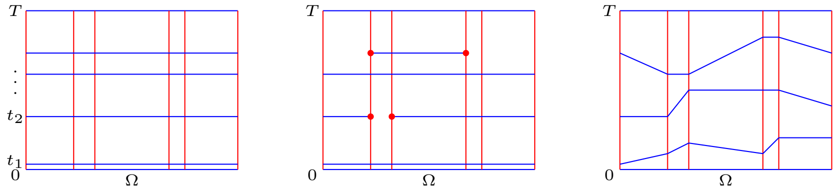

Figure 8: Space-time meshes for 1d spatial domains. Cartesian mesh without hanging nodes (left), mesh with hanging nodes (middle), non-Cartesian mesh (right)

(Definition 2.34) as a *marking* strategy.
As a *refinement* procedure, we use bisection of those time intervals which are marked.
The adaptive cycle is repeated until either a maximal number of time discretization points is reached or the error indicator η_y in (3.25) is below a prescribed tolerance.

Algorithm 2 Space-time finite element algorithm with time adaptivity for the state.

```
Input: Initial space-time mesh $\mathscr{T}_h^{init}$, tolerance $\varepsilon > 0$, refinement parameter $\theta \in (0,1)$, desired
    state $y_d$, initial state $y_0$, regularization parameter $\alpha > 0$, initial number of time discretization
    points $n$, maximal number of time points $n_{\max}$.
Output: Time-adapted space-time mesh $\mathscr{T}_h$, state solution $y_h$.
Set $\mathscr{T}_h = \mathscr{T}_h^{init}$ with $n$ time points.
while $n < n_{\max}$ OR $\eta_y \geq \varepsilon$ do
    Define $\mathscr{V}_h$ as the finite element space corresponding to $\mathscr{T}_h$ according to (3.30).
    Solve (3.31).
    Estimate the error contributions from (3.25).
    Mark time intervals according to the Dörfler criterion (Definition 2.34) with parameter $\theta$.
    Refine the marked time intervals using bisection.
end while
```

Remark 3.8. *The space-time mixed finite element discretization for the adjoint variable follows along the lines. We define the function spaces*

$$\tilde{Y} := \{p \in H^1(0,T;H_0^1(\Omega)) : p(T) = 0 \ in \ \Omega\}, \ \tilde{W} := L^2(0,T;H^1(\Omega))$$

and introduce the auxiliary variable $\vartheta = -\Delta p$. The resulting mixed finite element formulation for the adjoint state reads as: find $p_h \in \tilde{Y}_{h,0}, \vartheta_h \in \mathscr{V}_h$ and $\vartheta_h|_{\partial\Omega} = -\tilde{y}_d$ such that

$$\begin{aligned}\int_0^T \int_\Omega (p_h)_t v_t + \frac{1}{\alpha} p_h \, v + \nabla\vartheta_h \nabla v dx dt + \int_\Omega \nabla p_h(0) \nabla v(0) dx \\ = \int_0^T \int_\Omega (-(y_d)_t + \Delta y_d) v \, dx dt - \int_\Omega (y_d(0) - y_0) v(0) dx \quad \forall v \in \tilde{Y}_{h,0}, \qquad (3.33a)\end{aligned}$$

$$\int_0^T \int_\Omega -\nabla p_h \nabla\phi + \vartheta_h \phi \, dx dt = 0 \quad \forall \phi \in W_{h,0}, \qquad (3.33b)$$

where $\tilde{Y}_{h,0} = \mathscr{V}_h \cap \tilde{Y}$ and $\tilde{y}_d \in \mathscr{V}_h$ is an approximation of y_d. For more details we refer to [82, Section 4].

The space-time finite element algorithm with time adaptivity for the adjoint variable is summarized in Algorithm 3.

Algorithm 3 Space-time finite element algorithm with time adaptivity for the adjoint state.

```
Input: Initial space-time mesh $\mathscr{T}_h^{init}$, tolerance $\varepsilon > 0$, refinement parameter $\theta \in (0,1)$, desired
    state $y_d$, initial state $y_0$, regularization parameter $\alpha > 0$, initial number of time discretization
    points $n$, maximal number of time points $n_{\max}$.
Output: Time adapted space-time mesh $\mathscr{T}_h$, adjoint solution $p_h$.
Set $\mathscr{T}_h = \mathscr{T}_h^{init}$ with $n$ time points.
while $n < n_{\max}$ OR $\eta_p \geq \varepsilon$ do
    Define $\mathscr{V}_h$ as the finite element space corresponding to $\mathscr{T}_h$ according to (3.30).
    Solve (3.33).
    Estimate the error contributions from (3.27).
    Mark time intervals according to the Dörfler criterion (Definition 2.34) with parameter $\theta$.
    Refine the marked time intervals using bisection.
end while
```

3.6 Snapshot location in optimal control for POD model order reduction

Let us recall the POD method from Section 2.4. The POD method is built on an optimal (with respect to the chosen inner product) orthonormal least-squares approximation of a given solution trajectory $y(t), t \in [0,T]$ by means of a few POD basis functions $\psi_1, \ldots, \psi_\ell$, see (2.5). In practice, we do not have the whole solution trajectory at hand, but only a finite number of approximations at some time instances. Hence, we consider the time-discrete POD problem (2.7), which can be interpreted as a time-discrete approximation of (2.5) with an appropriate choice for the weights $\{\alpha_j\}_{j=0}^n$. A fundamental property of the POD basis is its strong dependence on the snapshots. In [128] this is described in the principle *"If it is not in the snapshots, it is not in the ROM"* or equivalently, *"you can only describe what you've seen already"* in [105, Section 3.3.1], meaning that a POD reduced-order model can only ever be as good as the snapshots. If a dynamical feature is missing in the snapshot ensemble, it will be missing in the POD reduced model, if the reduced model is not updated somehow.

We focus on simulation based POD model order reduction for optimal control problems, i.e. we generate snapshots by solving the state and the adjoint equation. In order to compute snapshots which reflect the dynamics of the optimal time-continuous trajectories $y(t)$ and $p(t)$ respectively $u(t)$ sufficiently well, we need to decide on

(i) a suitable spatial discretization which resolves the dynamical distribution in the spatial domain sufficiently well,

(ii) a suitable temporal grid at which snapshots are located in order to capture important dynamical properties within the time domain,

(iii) a suitable input control for snapshot generation.

The first issue (i) will be investigated in Section 4. In this section, we focus on a uniform spatial discretization. We address issues (ii) and (iii) in this section in the following way. In Section 3.6.1 the snapshot location methodology is presented and discussed. An error analysis of the proposed approach is given in Section 3.6.2. In Section 3.6.3, a numerical example is presented.

3.6.1 Methodology

The strategy of finding suitable time locations in order to generate the snapshots is based on (3.25) and (3.27), respectively. The basic idea is to solve either of the elliptic systems (3.17) and (3.21) adaptively with respect to time using the a-posteriori error estimates according to Theorem 3.6 and Theorem 3.7, respectively. The resulting time instances are used as locations for snapshot generation. At the same time, we aim to keep the computational effort to find these snapshot locations low. In particular, the POD reduced-order modeling of the optimal control problem including offline costs for snapshot location and POD basis computation shall give a notable speed up in comparison to solving the full-order model. For this reason, a coarse spatial resolution is used for solving the elliptic systems (3.17) and (3.21). This will be heuristically justified in the numerical tests where it turns out that spatial and temporal discretization decouple for the considered problem setting and data.

Besides the computation of suitable time instances for snapshot generation, the proposed method produces an approximation of the optimal control at the same time which serves as input control for snapshot generation.

Moreover, it turns out that it is beneficial to use these snapshot locations also as time grid points in the POD reduced optimization process, since they are related to the optimal solution.

We investigate three possible approaches:

(i) *Snapshot location tailored for the state.* We solve the elliptic system (3.17) adaptively with respect to time using the error estimator (3.25) according to Algorithm 2. This computation is done using a coarse spatial resolution in order to keep the computational costs low. As a result, we get time points which are related to the optimal state variable. At the same time an approximation of the optimal state is computed. The solution of the adjoint equation (3.10) on a coarse spatial resolution with the approximate state as right-hand side delivers an approximate adjoint state. From the optimality condition (3.11) we get an approximation of the optimal control with respect to a coarse spatial resolution which is utilized as input control for snapshot generation. A summary of this approach is given by Algorithm 4.

(ii) *Snapshot location tailored for the adjoint state / control.* We solve the elliptic system (3.21) adaptively with respect to time using the error estimator (3.27) on a coarse spatial grid. As a result, we obtain time grid points which are related to the optimal adjoint state. At the same time, an approximation of the optimal adjoint state is computed. According to the optimality condition (3.11) we compute an approximation of the optimal control with respect to a coarse spatial resolution. This control is then utilized as input control for snapshot generation. A summary of this approach is given by Algorithm 5.

(iii) *Snapshot location tailored for both state and adjoint state / control.* We run both strategies (i) and (ii) and then use the union of the time grid points as snapshot locations. At the same time, we compute two approximations of the optimal control on a coarse spatial mesh at different time points. From these, we compute a joint approximation as input control for snapshot generation. This approach is summarized in Algorithm 6.

Note that by Algorithm 2 and Algorithm 3 we obtain the state and adjoint solution to the biharmonic problem, respectively, which live in a space-time finite element space, i.e. $y_h \in Y_h$ and $p_h \in \tilde{Y}_{h,0}$. In order to compute an approximation for the optimal control which lives for each time instance in the finite element space V_h, we need to utilize an interpolation. This is possible due to the specific construction of the triangulation into space-time slabs $(t_{j-1}, t_j) \times \Omega$. In particular, we use the Lagrange interpolation π in order to interpolate the space-time discrete functions $y_h \in Y_h$ and $p_h \in \tilde{Y}_{h,0}$ for each time instance t_j into the finite element space V_h, i.e. $\pi(y_h(t_j)), \pi(p_h(t_j)) \in V_h$.

Discussion

We discuss the approaches (i), (ii), (iii) with regard to the suitability for the optimization interest, the computational efforts and the heuristic flavor.

Suitability for the optimization interest. If one expects significant variations of the dynamics

Algorithm 4 Adaptive snapshot location selection for linear-quadratic optimal control problems (tailored for the state).

```
Input: Coarse uniform spatial grid with resolution $h^+$.
Output: Snapshot locations $\{t_j\}_{j=0}^n$, approximation of the optimal control $u_{h^+}$.
1: Call Algorithm 2 in order to solve (3.17) adaptively w.r.t. time using the error estimator (3.25) with spatial resolution $h^+$. Obtain time instances $\{t_j\}_{j=0}^n = \mathcal{I}$ and the solution $y_{h^+} \in Y_h$.
2: Solve (3.10) on $\mathcal{I}$ with uniform spatial resolution $h^+$ and right-hand side $\pi y_{h^+}$ to obtain $p_{h^+}$.
3: Set $u_{h^+} = -\frac{1}{\alpha} p_{h^+}$.
```

Algorithm 5 Adaptive snapshot location selection for linear-quadratic optimal control problems (tailored for the adjoint state / control).

```
Input: Coarse uniform spatial grid with resolution $h^+$.
Output: Snapshot locations $\{t_j\}_{j=0}^n$, approximation of the optimal control $u_{h^+}$.
1: Call Algorithm 3 in order to solve (3.21) adaptively w.r.t. time using the error estimator (3.27) with spatial resolution $h^+$. Obtain time instances $\{t_j\}_{j=0}^n = \mathcal{I}$ and the solution $p_{h^+}$.
2: Set $u_{h^+} = -\frac{1}{\alpha} \pi p_{h^+}$.
```

in time in either of the state or control variable while the other variable is expected to behave mild, then an adaptive snapshot location strategy which is tailored for either the state or control might be advantageous. However, if the dynamical behavior of both state and adjoint state differs significantly over time and strong variations in the temporal domain are expected, the resulting time grids from Algorithms 4 and 5 might also differ strongly. Thus, the adaptive time grid tailored for the state variable constructed according to Algorithm 4 might not be a suitable time grid for the adjoint variable and vice versa. Only Algorithm 6 ensures to capture both dynamical evolutions of the state and adjoint state over time. However, if one expects a similar temporal behavior of the state and adjoint variables, it might suffice to only consider one of the Algorithms 4 and 5 depending on the quantity of interest and the application.

Computational efforts. Concerning the snapshot location computation, Algorithms 4 and 5 have the same computational effort. However, the computation of the approximate control is cheaper in Algorithm 5 since it only requires the solution of (3.11), while in Algorithm 4 it requires additionally one solve of (3.10). The approach (iii) is the most expensive approach. The computational costs are the sum of the computational costs of Algorithms 4 and 5 plus the additional costs for the construction of the approximate control u_{h^+} according to step 3 in Algorithm 6.

Heuristic flavor. The efficiency of the proposed approaches is based on the assumption that

Algorithm 6 Adaptive snapshot location selection for linear-quadratic optimal control problems (tailored for both state and adjoint state / control).

```
Input: Coarse uniform spatial grid with resolution $h^+$.
Output: Snapshot locations $\{t_j\}_{j=0}^n$, approximation of the optimal control $u_{h^+}$.
1: Run Algorithm 4 and obtain snapshot locations $\{t_j^1\}_{j=0}^{n^1}$ and an approximation of the optimal control $u_{h^+}^1$.
2: Run Algorithm 5 and obtain snapshot locations $\{t_j^2\}_{j=0}^{n^2}$ and an approximation of the optimal control $u_{h^+}^2$.
3: Set $\{t_j\}_{j=0}^n = \{t_j^1\}_{j=0}^{n^1} \cup \{t_j^2\}_{j=0}^{n^2}$ and set for $j = 0, \dots, n$
   $$u_{h^+}(t_j) = \begin{cases} u_{h^+}^1(t_j) & \text{if } t_j \in \{t_j^1\}_{j=0}^{n^1} \text{ and } t_j \notin \{t_j^2\}_{j=0}^{n^2}, \\ u_{h^+}^2(t_j) & \text{if } t_j \in \{t_j^2\}_{j=0}^{n^2} \text{ and } t_j \notin \{t_j^1\}_{j=0}^{n^1}, \\ \frac{1}{2}(u_{h^+}^1(t_j) + u_{h^+}^2(t_j)) & \text{if } t_j \in \{t_j^1\}_{j=0}^{n^1} \text{ and } t_j \in \{t_j^2\}_{j=0}^{n^2}. \end{cases}$$
```

temporal and spatial discretization decouple for the considered problem setting and data. This behavior is discovered in numerical experiments. In this way, the method has a certain heuristic flavor. Based on this observation we use a coarse spatial resolution in the offline phase in order to generate suitable snapshot locations. In general, the issue of when spatial and temporal discretization decouple is not easy to answer. Related to this issue, we refer to [178, Chapter 3], where the relationship of temporal and spatial discretization is analyzed in the context of unsteady internal flow. In [30] a combination of time adaptivity and spatial adaptivity is considered including a-priori and a-posteriori error analysis.

POD reduced-order modeling for the optimal control problem

After the computation of the snapshot locations and an approximate control u_{h^+} using either of the Algorithms 4 to 6, the usual POD offline phase begins. For this, we sample state and adjoint snapshots at the computed snapshot locations with respect to a fine spatial resolution by solving (3.15) with control input $\mathbb{I}u_{h^+}$ and (3.16), where $\mathbb{I}$ denotes a suitable interpolation of the control u_{h^+} from a coarse spatial grid onto the fine finite element grid.
In order to derive a POD reduction of the optimal control problem (3.9), we follow the *first-reduce-then-optimize* approach, see Section 2.4.2. We utilize an *integrated* snapshot ensemble, i.e. we take into account both state and adjoint snapshots, compare Remark 2.47. In this way, the POD projection error for both state and adjoint variables is determined by the sum of the neglected eigenvalues. Assume we have constructed a POD reduced space $X_\ell \subset H_0^1(\Omega)$ according to (2.7). Let us denote by $\tilde{y}_0$ a suitable approximation of the initial value y_0 in the reduced space X_ℓ. For example, it can be computed as the solution to

$$(\tilde{y}_0, \psi)_{L^2(\Omega)} = (y_0, \psi)_{L^2(\Omega)} \quad \forall \psi \in X_\ell.$$

Note that it is also possible to utilize a homogenization with respect to the initial condition (according to e.g. [89, Remark 3.3]) which then allows to construct a reduced POD model for which the POD reduced state fulfills the initial condition exactly, see [89, Section 3.3]. The POD reduced-order optimal control problem reads as follows:

$$\min_{(y_\ell, u_\ell) \in W(0,T;X_\ell) \times L^2(0,T;\Omega)} J_\ell(y_\ell, u_\ell) \quad \text{s.t.} \quad y_\ell = \mathcal{S}_\ell(u_\ell, \tilde{y}_0), \tag{3.34}$$

where the linear operator

$$\mathcal{S}_\ell : L^2(0,T;H^{-1}(\Omega)) \times X_\ell \to W(0,T;X_\ell), \quad (\hat{u}, \tilde{y}_0) \mapsto y_\ell := \mathcal{S}_\ell(\hat{u}, \tilde{y}_0)$$

is defined analogously to the solution operator $\mathcal{S}$ defined in (3.5). In particular, we replace the space $H_0^1(\Omega)$ in (3.2) by X_ℓ. This leads to the following reduced-order state equation: for a given initial condition $y_\ell(0) := \tilde{y}_0$ determine a function $y_\ell \in W(0,T;X_\ell)$ such that

$$\frac{d}{dt}(y_\ell(t), \psi)_{L^2(\Omega)} + (\nabla y_\ell(t), \nabla \psi)_{L^2(\Omega)} = (u_\ell(t), \psi)_{L^2(\Omega)} \quad \text{for } t \in [0,T], \psi \in X_\ell. \tag{3.35}$$

Note that we use the notation u_ℓ for the control in order to indicate the affiliation of the control to the reduced-order optimization problem. This helps to distinguish notationally between the solution u to (3.9) and the solution u_ℓ to (3.34). However, note carefully that we do not impose a POD reduction for the control space. In order to achieve a conservative reduction of the control variable, the reduction of the control is induced by the reduction of the adjoint variable due to the optimality condition

$$\alpha u_\ell + p_\ell = 0 \quad \text{in } [0,T] \times \Omega, \tag{3.36}$$

where p_ℓ denotes the solution to the following reduced-order adjoint equation: determine a function $p_\ell \in W(0,T;X_\ell)$ such that $p_\ell(T) = 0$ and

$$-\frac{d}{dt}(p_\ell(t), \psi)_{L^2(\Omega)} + (\nabla p_\ell(t), \nabla \psi)_{L^2(\Omega)} = ((y_\ell - y_d)(t), \psi)_{L^2(\Omega)} \quad \text{for } t \in [0,T], \psi \in X_\ell, \tag{3.37a}$$

Existence of a unique solution y_ℓ and p_ℓ to (3.35) and (3.37), respectively, as well as stability results follow analogously to the infinite-dimensional case according to [69] and $X_\ell \subset H_0^1(\Omega)$.

Now, the POD online phase begins in which (3.34) is solved. Since we already have a time grid at hand which is related to the optimal control problem, it makes sense to use this time grid for solving the problem (3.34). The POD reduced-order modeling for the linear-quadratic optimal control problems using adaptive snapshot computation is summarized in Algorithm 7.

Algorithm 7 POD model order reduction for linear-quadratic optimal control problems using adaptive snapshot locations.

```
Input: Snapshot locations $\{t_j\}_{j=0}^n = \mathcal{I}$ and input control $u_{h^+}$ from either of the Algorithms 4
    to 6, fine spatial resolution $h^*$, number of POD basis functions $\ell$
Output: $y_\ell, u_\ell$
  Sample state and adjoint snapshots in a simulation of (3.15) and (3.16) with spatial resolution
    $h^*$ and input control $\mathbb{I}u_{h^+}$ on $\mathcal{I}$.
  Compute a POD basis of rank $\ell$ according to (2.7) using an integrated snapshot ensemble.
  Set up and solve the POD-ROM (3.34) on $\mathcal{I}$ to obtain $y_\ell, u_\ell$.
```

Remark 3.9. *In principle, it is conceivable to derive a POD reduced-order formulation for the biharmonic systems* (3.17) *and* (3.21)*, respectively. However, this approach is not followed here, since our motivation is to use the usual optimization algorithm, for example of Remark 2.45, and utilize existing implementations. The snapshot location strategy is considered as an add-on with the goal to deliver time grid points and an input control which are associated with the optimal solution. For a space-time POD Galerkin approach we refer to [26, 187].*

3.6.2 Error analysis

We want to know how well the time-discrete solution to the POD reduced-order optimization problem (3.34) approximates the optimal solution to (3.9).
We use the following notation. Suppose that $\ell \in \mathbb{N}$ is fixed. Let $\bar{u}$ be the optimal control, i.e. the solution to (3.9) and let $\bar{u}_\ell$ denote the optimal POD control, i.e. the solution to (3.34). Let $\bar{y} = y(\bar{u})$ denote the associated state and $\bar{p} = p(\bar{y}) = p(y(\bar{u}))$ is the associated adjoint state. Analogously, let $\bar{y}_\ell = y_\ell(\bar{u}_\ell)$ denote the associated POD state and $\bar{p}_\ell = p_\ell(\bar{y}_\ell) = p_\ell(y_\ell(\bar{u}_\ell))$ the associated adjoint state. We denote by y_k and p_k the solution to the time-discrete problem (3.24) and (3.26), respectively. By $\tilde{y}_k(u_p)$ and $\tilde{p}_k(u_p) = \tilde{p}_k(\tilde{y}_k(u_p))$ we denote time-discrete solutions to (3.2) and (3.12), respectively, for a given particular control u_p. Moreover, we define the time-discrete controls $u_k := -\frac{1}{\alpha}p_k$ and $\hat{u}_k := -\frac{1}{\alpha}\tilde{p}_k(y_k)$. We make the assumptions

$$\|y_k - \tilde{y}_k(\hat{u}_k)\| < \varepsilon, \tag{3.38a}$$

$$\|p_k - \tilde{p}_k(\tilde{y}_k(u_k))\| < \varepsilon, \tag{3.38b}$$

for $0 < \varepsilon \ll 1$. By $(\bar{y}_k^\ell, \bar{u}_k^\ell)$ we denote the time-discrete solution to (3.34) and $\bar{p}_k^\ell$ is the associated adjoint state.

For clarity, we indicate by $\|\cdot\|$ the $\|\cdot\|_{L^2(0,T;X)}$-norm for either $X = V = H_0^1(\Omega)$ or $X = H = L^2(\Omega)$. We define the sets H_ℓ and V_ℓ by $H_\ell = \text{span}\{\psi_1^H, \dots, \psi_\ell^H\} \subset V \subset H$ and $V_\ell = \text{span}\{\psi_1^V, \dots, \psi_\ell^V\} \subset V$, where the superscript indicates which space is used for X in (2.5). We introduce the orthogonal projection $\mathcal{P}^\ell$, which can either be $\mathcal{P}_H^\ell : H \to H_\ell \subset V$ or $\mathcal{P}_V^\ell : V \to V_\ell \subset V$ as follows

$$\text{for } f \in H, \quad \mathcal{P}_H^\ell f \quad \text{minimizes} \quad \inf_{f_\ell \in H_\ell} \|f - f_\ell\|_H$$

and

$$\text{for } f \in V, \quad \mathcal{P}^\ell_V f \quad \text{minimizes} \quad \inf_{f_\ell \in V_\ell} \|f - f_\ell\|_V.$$

These projections can be computed using the following expressions

$$\mathcal{P}^\ell_H f = \sum_{i=1}^{\ell} (f, \psi_i^H)_H \psi_i^H \quad \text{and} \quad \mathcal{P}^\ell_V f = \sum_{i=1}^{\ell} (f, \psi_i^V)_V \psi_i^V.$$

Error bound for the state

Let us start with the error estimation with respect to the state variable. We are interested in estimating the quantity

$$\|\bar{y} - \tilde{y}^\ell_k\|.$$

Using the null-trick and applying the triangle inequality we arrive at

$$\|\bar{y} - \tilde{y}^\ell_k\| \leq \underbrace{\|\bar{y} - y_k\|}_{(3.39.i)} + \underbrace{\|y_k - \tilde{y}_k(\hat{u}_k)\|}_{(3.39.ii)} + \underbrace{\|\tilde{y}_k(\hat{u}_k) - \tilde{y}^\ell_k(\hat{u}_k)\|}_{(3.39.iii)} + \underbrace{\|\tilde{y}^\ell_k(\hat{u}_k) - \tilde{y}^\ell_k\|}_{(3.39.iv)} \tag{3.39}$$

Now, we analyze and estimate each term in (3.39).

If we compute the snapshot locations and the input control with either of the Algorithms 4 and 6, then the term (3.39.i) can be estimated utilizing the a-posteriori error estimate (3.25) for the state and concerns the snapshot location. Since $V \hookrightarrow H$ there exists an embedding constant c_v such that

$$\|v\|_H \leq c_v \|v\|_V \quad \forall v \in V.$$

In particular, it holds

$$\|\bar{y} - y_k\|^2_{L^2(0,T;H)} \leq c_v \|\bar{y} - y_k\|^2_{L^2(0,T;V)} \leq c_v \|\bar{y} - y_k\|^2_{H^{2,1}(0,T;\Omega)} \leq c_1 \, \eta_y^2, \tag{3.40}$$

where $c_1 := c_v \cdot c$ with c denoting the constant from (3.25). The use of sufficiently enough time instances ensures that the indicator η_y drops below a prescribed tolerance.

For the term (3.39.ii) we make use of the assumption (3.38a).

Let us now focus on the part (3.39.iii). For this, we make use of the decomposition

$$\tilde{y}_k(\hat{u}_k) - \tilde{y}^\ell_k(\hat{u}_k) = \tilde{y}_k(\hat{u}_k) - \mathcal{P}^\ell \tilde{y}_k(\hat{u}_k) + \mathcal{P}^\ell \tilde{y}_k(\hat{u}_k) - \tilde{y}^\ell_k(\hat{u}_k).$$

After the computation of the time-discrete solution y_k of the biharmonic equation, we take the control $\hat{u}_k = -\frac{1}{\alpha}\tilde{p}_k(y_k)$ as input control for snapshot generation. Since we utilize an integrated snapshot set, i.e. we utilize both state and adjoint information, we consider the POD problem

$$\min_{\psi_1,\ldots,\psi_\ell \in X} \int_0^T \left(\|\tilde{y}_k(t) - \sum_{i=1}^{\ell} (\tilde{y}_k(t), \psi_i)_X \psi_i\|^2_X + \|\tilde{p}_k(t) - \sum_{i=1}^{\ell} (\tilde{p}_k(t), \psi_i)_X \psi_i\|^2_X \right) dt = \sum_{i>\ell} \lambda_i,$$

where we left out the dependency of $\tilde{y}_k = \tilde{y}_k(\hat{u}_k)$ and $\tilde{p}_k = \tilde{p}_k(y_k)$ for the sake of clarity. This gives us the estimation

$$\|\tilde{y}_k(\hat{u}_k) - \mathcal{P}^\ell \tilde{y}_k(\hat{u}_k)\|^2 \leq \sum_{i>\ell} \lambda_i. \tag{3.41}$$

What is left, is to estimate the quantity $\mathcal{P}^\ell \tilde{y}_k(\hat{u}_k) - \tilde{y}^\ell_k(\hat{u}_k) =: \vartheta$. For this, we follow [103, Proposition 4.7, equation (4.19)] which gives the estimate

$$\operatorname*{ess\,sup}_{t\in[0,T]} \|\vartheta(t)\|^2_H + \int_0^T \|\vartheta(t)\|^2_V dt \leq \|\vartheta(0)\|^2_H + \|(\tilde{y}_k(\hat{u}_k))_t - \mathcal{P}^\ell(\tilde{y}_k(\hat{u}_k))_t\|^2_{L^2(0,T;V^*)}. \tag{3.42}$$

Finally, we consider the last part (3.39.iv). Deriving an energy estimate for the POD reduced state equation following (3.4) leads to the estimate

$$\|\tilde{y}_k^\ell(\hat{u}_k) - \bar{y}_k^\ell\|_{L^2(0,T;H)} \leq c_v \|\tilde{y}_k^\ell(\hat{u}_k) - \bar{y}_k^\ell\|_{L^2(0,T;V)} \leq c_2 \|\hat{u}_k - \bar{u}_k^\ell\|_{L^2(0,T;\Omega)},$$

with $c_2 = c_v \cdot c$ and c according to (3.4). By $\bar{u}_k^\ell$ we denote the time-discrete optimal control to the POD reduced optimal control problem with POD basis computed from snapshots generated from an input control $\hat{u}_k$. In order to control the quantity

$$\|\hat{u}_k - \bar{u}_k^\ell\|_{L^2(0,T;\Omega)} \leq \|\hat{u}_k - \bar{u}\|_{L^2(0,T;\Omega)} + \|\bar{u} - \bar{u}_k^\ell\|_{L^2(0,T;\Omega)}$$

we make use of the a-posteriori error estimation according to [177] which provides an upper bound for the error between the (unknown) optimal control $\bar{u}$ and an particular control u_p (here $u_p = \hat{u}_k$ and $u_p = \bar{u}_k^\ell$). It holds

$$\|\bar{u} - u_p\|_{L^2(0,T;\Omega)} \leq \frac{1}{\alpha} \|\zeta_p\|_{L^2(0,T;\Omega)}, \tag{3.43}$$

according to [177, Theorem 3.1] adapted to this setting. The perturbation $\zeta_p \in L^2(0,T;\Omega)$ is chosen such that $\alpha u_p + p_p + \zeta_p = 0$ holds, where p_p denotes the adjoint state associated with the control u_p. For the construction of ζ_p we refer to [177, Proposition 3.2] which involves a full evaluation of the adjoint and state equation (and not a solution of their POD approximations).

In summary, we can build the following theorem.

Theorem 3.10. *Let $\bar{y}$ denote the optimal state for (3.9) and let $\bar{y}_k^\ell$ denote the time-discrete solution to (3.34). The POD basis is computed from an integrated snapshot set with input control and snapshot locations computed according to either of the Algorithms 4 and 6. In (2.5) it is either $X = V$ or $X = H$. Then, it holds true*

$$\begin{aligned}\|\bar{y} - \bar{y}_k^\ell\|_{L^2(0,T;X)} \leq \sqrt{c_1}\eta_y + \varepsilon + \left(\sum_{i>\ell} \lambda_i\right)^{1/2} &+ \|(\tilde{y}_k(\hat{u}_k))_t - \mathcal{P}^\ell(\tilde{y}_k(\hat{u}_k))_t\|_{L^2(0,T;V^*)} \\ &+ \frac{c_2}{\alpha}(\|\zeta_1\|_{L^2(0,T;\Omega)} + \|\zeta_2\|_{L^2(0,T;\Omega)}),\end{aligned} \tag{3.44}$$

where $\zeta_1, \zeta_2 \in L^2(0,T;\Omega)$ are chosen such that $\alpha\hat{u}_k + \tilde{p}_k(\hat{u}_k) + \zeta_1 = 0$ and $\alpha\bar{u}_k^\ell + \tilde{p}_k(\bar{u}_k^\ell) + \zeta_2 = 0$, respectively, is fulfilled and $c_1, c_2 > 0$ are constants.

Proof. Utilizing the decomposition (3.39) together with (3.40), (3.38a), (3.41), (3.42) and (3.43) leads to the claim. □

Remark 3.11. *i) Note that the error estimation (3.44) in Theorem 3.10 depends on an error of time derivatives of the state. As discussed in [103, Remark 4.8(b)] this dependency can be avoided by enriching the snapshot set with time derivatives (more specifically, finite difference approximations of time derivatives), see also [102, 121].*
Note that for a specific setting it is possible to avoid to include time derivatives into the snapshot set. This is based on [165]. A comprehensive analysis on whether snapshot difference quotients are needed in POD is given in [109].

ii) Note that Theorem 3.10 is valid if the snapshot location strategy of either of the Algorithms 4 and 6, respectively, is used, i.e. the snapshot locations are related to the optimal state. However, if Algorithm 5 is utilized (snapshot locations are only related to the adjoint state), it is not guaranteed that the error contribution (3.39.i) is small, i.e. the selected time grid points which are related to the optimal adjoint state might not be suitable for the state variable. In Section 3.8 it is not possible to reformulate the optimality system as a biharmonic equation for the state variable. For this case, a post-processing step is introduced in Section 3.8.2.

Error bound for the control

We are interested in estimating how well the optimal control $\bar{u}$ is approximated by the time-discrete POD reduced-order solution $\bar{u}_k^\ell$. For this, the a-posteriori error estimate according to [177] can be utilized. Moreover, we recall the optimality condition (3.11) and a time discretization of the optimality condition (3.36), i.e.

$$\begin{aligned} \alpha\bar{u} + \bar{p} &= 0 \quad \text{in } [0,T]\times\Omega, \\ \alpha\bar{u}_k^\ell + \bar{p}_k^\ell &= 0 \quad \text{in } [0,T]\times\Omega. \end{aligned}$$

Subtracting both equations leads to

$$\bar{u} - \bar{u}_k^\ell = -\frac{1}{\alpha}(\bar{p} - \bar{p}_k^\ell) \quad \text{in } [0,T]\times\Omega$$

and thus

$$\|\bar{u} - \bar{u}_k^\ell\|_{L^2(0,T;X)} = \frac{1}{\alpha}\|\bar{p} - \bar{p}_k^\ell\|_{L^2(0,T;X)} \quad \text{in } [0,T]\times\Omega. \tag{3.45}$$

Therefore, we now consider an error bound for the adjoint variable.

Error bound for the adjoint state

Analogously to the error analysis for the state, we derive an error estimation for the quantity

$$\|\bar{p} - \bar{p}_k^\ell\|.$$

This is done along the lines to the error bound for the state, so we shorten the analysis to the main points. Using the null-trick and applying the triangle inequality we arrive at

$$\|\bar{p} - \bar{p}_k^\ell\| \le \underbrace{\|\bar{p} - p_k\|}_{(3.46.i)} + \underbrace{\|p_k - \tilde{p}_k(u_k)\|}_{(3.46.ii)} + \underbrace{\|\tilde{p}_k(u_k) - \tilde{p}_k^\ell(u_k)\|}_{(3.46.iii)} + \underbrace{\|\tilde{p}_k^\ell(u_k) - \bar{p}_k^\ell\|}_{(3.46.iv)} \tag{3.46}$$

If we compute the snapshot locations and the input control with Algorithm 5, then the term (3.46.i) can be estimated utilizing the a-posteriori error estimate (3.27) for the adjoint state and concerns the snapshot location. This yields to

$$\|\bar{p} - p_k\|^2 \le c_v\|\bar{p} - p_k\|^2_{H^{2,1}(0,T;\Omega)} \le c_1\eta_p^2. \tag{3.47}$$

For term (3.46.ii) we utilize (3.38b). In order to estimate (3.46.iii), we make use of the decomposition

$$\tilde{p}_k(u_k) - \tilde{p}_k^\ell(u_k) = \tilde{p}_k(u_k) - \mathcal{P}^\ell\tilde{p}_k(u_k) + \mathcal{P}^\ell\tilde{p}_k(u_k) - \tilde{p}_k^\ell(u_k).$$

In line with (3.41), it holds

$$\|\tilde{p}_k(u_k) - \mathcal{P}^\ell\tilde{p}_k(u_k)\|^2 \le \sum_{i>\ell}\lambda_i. \tag{3.48}$$

In order to estimate the quantity $\mathcal{P}^\ell\tilde{p}_k(u_k) - \tilde{p}_k^\ell(u_k) =: \theta$, we get from [103, Proposition 4.7] the estimation

$$\begin{aligned} \operatorname*{ess\,sup}_{t\in[0,T]}\|\theta(t)\|_H^2 + \int_0^T \|\theta(s)\|_V^2 dt \le c\Bigg(&\sum_{i>\ell}\lambda_i + \|(\tilde{p}_k(u_k))_t - \mathcal{P}^\ell(\tilde{p}_k(u_k))_t\|^2_{L^2(0,T;V^*)} \\ &+ \|\tilde{y}_k(u_k) - \tilde{y}_k^\ell(u_k)\|^2_{L^2(0,T;H)}\Bigg). \end{aligned} \tag{3.49}$$

Finally, the last term (3.46.iv) can be estimated in terms of the control according to the energy estimate (3.14) for the POD reduced-order adjoint equation. Thus, it holds

$$\|\tilde{p}_k^\ell(u_k) - \tilde{p}_k^\ell(\bar{u}_k^\ell)\| \le c\|u_k - \bar{u}_k^\ell\|_{L^2(0,T;\Omega)},$$

with constant $c > 0$ from (3.14). Again, with the use of the a-posteriori error estimation according to [177] this term can be controlled.
We summarize the findings in the following theorem.

Theorem 3.12. *Let $\bar{p}$ denote the optimal adjoint state for (3.9) and let $\bar{p}_k^\ell$ denote the time-discrete adjoint solution to a POD approximation of (3.9). The POD basis is computed from an integrated snapshot set with input control and snapshot locations computed according to Algorithm 5. In (2.5) it is either $X = V$ or $X = H$. Then, it holds true*

$$\begin{aligned}\|\bar{p} - \bar{p}_k^\ell\|_{L^2(0,T;X)} \leq \sqrt{c_1}\eta_p + \varepsilon + \left(c_2 \sum_{i>\ell} \lambda_i\right)^{1/2} &+ c_2\|(\tilde{p}_k(u_k))_t - \mathcal{P}^\ell(\tilde{p}_k(u_k))_t\|^2_{L^2(0,T;V^*)} \\ &+ c_2\|\tilde{y}_k(u_k) - \tilde{y}_k^\ell(u_k)\|^2_{L^2(0,T;H)} + \frac{c_3}{\alpha}(\|\zeta_1\|_{L^2(0,T;\Omega)} + \|\zeta_2\|_{L^2(0,T;\Omega)}),\end{aligned} \tag{3.50}$$

where $\zeta_1, \zeta_2 \in L^2(0,T;\Omega)$ are chosen such that $\alpha u_k + \tilde{p}_k(u_k) + \zeta_1 = 0$ and $\alpha \bar{u}_k^\ell + \tilde{p}_k(\bar{u}_k^\ell) + \zeta_2 = 0$, respectively, is fulfilled and $c_1, c_2, c_3 > 0$ are constants.

Proof. Utilizing the decomposition (3.46) together with (3.47), (3.38b), (3.48), (3.49) and (3.43) lead to the claim. □

Remark 3.13. *i) Note that the error estimate (3.50) in Theorem 3.12 depends on the time derivatives of the adjoint state. The same arguments of Remark 3.11(i) apply here.*

ii) The estimate (3.50) in Theorem 3.12 is further dependent on the error between the state snapshots with input control u_k and the POD approximation with the same input control. In order to estimate this, we can utilize the estimations for (3.39.iii). However, it is not guaranteed that the snapshot locations generated by Algorithm 5 lead to snapshot locations which are suitable for the state as well, compare Remark 3.11(ii).

3.6.3 Numerical results

In this numerical test, we construct an example in such a way that we know the analytical solution to the optimal control problem (3.9). Similar examples are presented in [9, 10]. All coding is done in MATLAB R2018a. The time-adaptive solution of the biharmonic system is built upon the codes from [82].

Problem setting

As spatial domain, we consider the one-dimensional interval $\Omega = (0,1) \subset \mathbb{R}$. The end time is $T = 1$ and the regularization parameter is $\alpha = 1$. We introduce a forcing term $f \in L^2(0,T;\Omega)$ on the right-hand side of the state equation (4.40a) as follows

$$y_t - \Delta y = u + f \quad \text{in } (0,T] \times \Omega.$$

The remaining setting of the optimal control problem (3.9) stays the same. In order to reformulate the resulting optimality system with respect to the state variable we need to assume higher regularity for the forcing term, in particular $f \in L^2(0,T;H^1(\Omega)) \cap H^1(0,T;L^2(\Omega))$. Then, in the biharmonic system (3.17), the equations include terms involving the forcing term in the following way

$$-y_{tt} + \Delta^2 y + \frac{1}{\alpha} y = \frac{1}{\alpha} y_d - \Delta f - f_t \quad \text{in } (0,T) \times \Omega, \tag{3.51a}$$
$$y = 0 \quad \text{on } [0,T] \times \partial\Omega, \tag{3.51b}$$
$$-\Delta y = f \quad \text{on } [0,T] \times \partial\Omega, \tag{3.51c}$$
$$(y_t - \Delta y)(T,\cdot) = f(T,\cdot) \quad \text{in } \Omega, \tag{3.51d}$$
$$y(0,\cdot) = y_0 \quad \text{in } \Omega. \tag{3.51e}$$

The resulting elliptic problem for the adjoint variable includes the forcing term in the following way

$$-p_{tt} + \Delta^2 p + \frac{1}{\alpha} p = f - (y_d)_t + \Delta y_d \quad \text{in } (0,T) \times \Omega, \tag{3.52a}$$

$$p = 0 \quad \text{on } [0,T] \times \partial\Omega, \tag{3.52b}$$

$$\Delta p = y_d \quad \text{on } [0,T] \times \partial\Omega, \tag{3.52c}$$

$$(p_t + \Delta p)(0,\cdot) = y_d(0,\cdot) - y_0 \quad \text{in } \Omega, \tag{3.52d}$$

$$p(T,\cdot) = 0 \quad \text{in } \Omega. \tag{3.52e}$$

Note that the residual based error indicators of Theorems 3.6 and 3.7 contain the terms involving the forcing term accordingly.

We choose the optimal state as

$$\bar{y}(t,x) = \sin(\pi x)\text{atan}\left(\frac{t-1}{\varepsilon}\right).$$

For small values of $0 < \varepsilon \ll 1$, the state variable $\bar{y}$ develops a steep gradient for $t \to 1_+$, compare Figure 9, which shows the part which accounts for the steep gradient in the temporal domain. For the numerical test, we use $\varepsilon = 10^{-05}$.

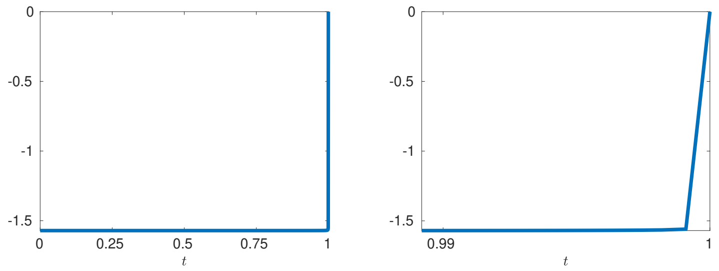

Figure 9: Time layer function atan $\left(\frac{t-1}{\varepsilon}\right)$ (left) and zoom in (right)

The optimal control is chosen as

$$\bar{u}(t,x) = -\sin(\pi x)\sin(\pi t).$$

The analytical optimal state, its contour lines and the optimal control is shown in Figure 10.

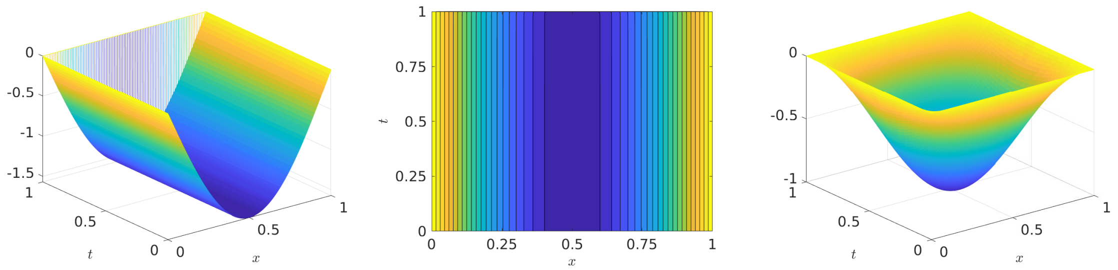

Figure 10: Analytical optimal state $\bar{y}$ (left), its contour lines (middle) and analytical optimal control $\bar{u}$

Accordingly, the initial condition is

$$y_0(x) = \sin(\pi x)\text{atan}\left(\frac{-1}{\varepsilon}\right)$$

and the desired state is

$$y_d(t,x) = \sin(\pi x)\left(\operatorname{atan}\left(\frac{t-1}{\varepsilon}\right) + \pi\cos(\pi t) - \pi^2\sin(\pi t)\right).$$

The forcing term is given as

$$f(t,x) = \sin(\pi x)\left(\frac{\varepsilon}{t^2-2t+\varepsilon^2+1} + \pi^2\operatorname{atan}\left(\frac{t-1}{\varepsilon}\right) + \sin(\pi t)\right).$$

In order to solve the optimal control problem (3.9) and the reduced-order optimal control problem (3.34), we use an implicit Euler method for temporal discretization, continuous piecewise linear finite elements for spatial discretization and a gradient method with Armijo step size determination, see Remark 2.45. As stopping criterion for the optimization, we use

$$\|\nabla\hat{J}_\ell(u)\| \le \varepsilon_r\|\nabla\hat{J}_\ell(0)\| + \varepsilon_a$$

or a maximal number `maxiter` of gradient iterations is reached, where we set $\varepsilon_r = 10^{-08}$ as relative tolerance, $\varepsilon_a = 10^{-08}$ as absolute tolerance and `maxiter` $= 20$.

As time weights for computing POD basis functions in (2.7), we use trapezoidal weights

$$\alpha_0 = \frac{\Delta t_1}{2}, \quad \alpha_j = \frac{\Delta t_{j+1}+\Delta t_j}{2} \text{ for } j = 1,\dots,n-1, \quad \alpha_n = \frac{\Delta t_n}{2}.$$

Moreover, in (2.7) we use the $X = L^2(\Omega)$ for the inner product and norm.

The focus in this numerical test example lies on snapshot location. Hence we compare a uniform placement of time instances for snapshot generation with an adaptive time discretization. So, for a fixed number of time instances $n+1$, we compare a uniform time grid with time step size $\Delta t = 1/n$ with an adaptive time grid with the same number of time points.
For the adaptive time discretization we compare the Algorithms 4 to 6. Due to the steep gradient in the optimal state at the end of the time interval we expect that a time grid which is adapted with respect to the state will improve the error in the state variable since we expect that it locates the time grid points towards the end of the time interval where the layer is located.

Uniform snapshot location

First, we consider POD reduced-order modeling with a uniform time discretization. For now, we consider an input control $u = 0$ for snapshot generation and focus on the influence of the snapshot locations and time discretization on the approximation quality. Later, we investigate other choices for the input control. We fix the POD basis length to $\ell = 1$ and investigate different lengths for the POD basis later. We utilize an integrated snapshot set (see Remark 2.47), i.e. we compute a POD basis corresponding to a union of state and adjoint snapshots. This POD basis is used in the POD Galerkin ansatz for both state and adjoint state. Figure 11 shows the solution y_ℓ to (3.34), its contour lines and the associated control u_ℓ. We choose $n = 27$, i.e. we have 28 time instances. We see that on a uniform grid, the layer in the optimal state at the end of the time interval is overshot, i.e. the approximation is not satisfactory.

Adaptive snapshot location

Now, we use the adaptive time selection according to Algorithms 4 to 6 and produce an adaptive time grid with the same number of time instances, i.e. $n = 27$, as in the uniform case. As initial time grid for the time-adaptive selection we start with a uniform time grid with 9 time instances. Then, according to the error indicator (3.25) and (3.27), respectively, suitable time instances are

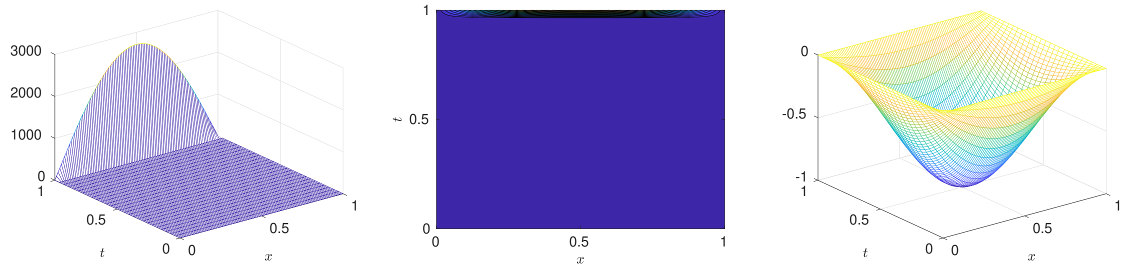

Figure 11: POD optimal state computed on an equidistant time grid with $n = 27$ (left), its contour lines (middle) and the POD optimal control (right)

selected. For the Dörfler criterion we use the parameter $\theta = 0.5$.

The resulting time grid for the adaptivity with respect to the state according to Algorithm 4 is shown in Figure 12 (left and middle) for different spatial resolutions. At the end of the time interval where the steep gradient is located, many time instances are located whereas before a coarse resolution is sufficient.
The resulting time grid for the adaptivity with respect to the adjoint according to Algorithm 5 is shown in Figure 13 (left and middle) for different spatial resolutions. The location of the time grid points is spread more since there are not as steep transitions as in the state variable.
The resulting time grid for the adaptivity addressing both state and adjoint state according to Algorithm 6 is shown in the right plot of Figure 13. Note that this joint time-adaptive grid has the same total number of time instances as in the pure state / adjoint state adaptive schemes in order to achieve a fair comparison. For completeness, wee see a uniform space-time discretization in the right plot of Figure 12.

We see for both state and adjoint variables in this example, that it does not matter whether a fine spatial resolution or a coarse spatial resolution is used, the resulting time grid is the exact same. This means that the temporal discretization in this case is insensitive against changes in the spatial resolution, i.e. spatial and temporal discretization decouple. This observation justifies the use of a coarse spatial resolution for the solution of the biharmonic system in the offline phase in order to achieve cheap computations.

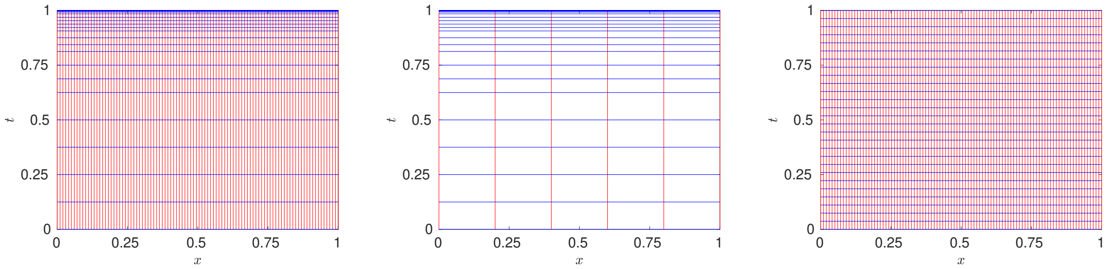

Figure 12: Adaptive space-time grids according to Algorithm 4 with $n = 27$ and $h^+ = 1/100$ (left) and $h^+ = 1/5$ (middle), respectively, and the equidistant grid with $n = 27$ and $h^+ = 1/100$ (right)

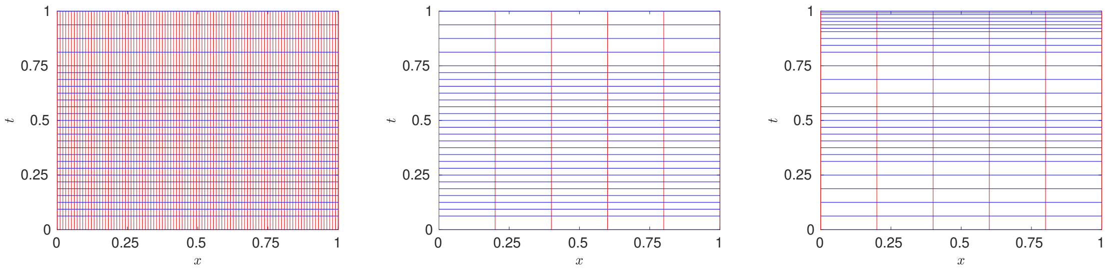

Figure 13: Adaptive space-time grids according to Algorithm 5 with $n = 27$ and $h^+ = 1/100$ (left) and $h^+ = 1/5$ (middle), respectively, and the adaptive space-time grid according to Algorithm 6 with $n = 27$ time points altogether (right)

POD model order reduction for optimal control using adaptive snapshot locations

Now we generate state and adjoint snapshots on the adaptive time mesh. Note that we now compute the snapshots corresponding to an input control $u = 0$ (in contrary to the Algorithms 4 to 6) in order to numerically compare the results to the uniform case, where the only difference between the models is the time discretization. A different input control is investigated later. We also solve the POD reduced-order model optimization problem on the time-adaptive mesh.
The POD optimal state, its contour lines and the POD optimal control utilizing the state adaptive grid according to Algorithm 4 are shown in Figure 14. Note that this looks more similar to the analytical solution shown in Figure 10 than the POD solution on a uniform time mesh. Especially note the different scaling. The layer is captured better since there are more time instances located at the end of the time interval in comparison to a uniform discretization.

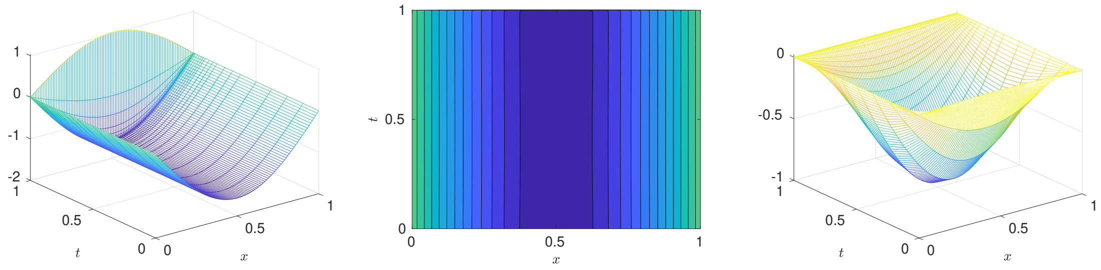

Figure 14: POD optimal state computed on the adaptive time grid with $n = 27$ (left), its contour lines (middle) and the POD optimal control using Algorithm 4 with input control $u = 0$ (right)

The POD optimal state, its contour lines and the POD optimal control utilizing the adaptive grid regarding the adjoint state according to Algorithm 5 are shown in Figure 15. We observe that the results for the state solution look even worse than using a uniform time mesh, since the time discretization at the end of the time interval is coarser in the case of the adjoint adapted mesh than in the uniform case. In particular, the adaptive time grid for the adjoint state turns out to be not suitable for the approximation of the state.
Finally, the POD optimal state, its contour lines and the POD optimal control utilizing the union of the adaptive grids according to Algorithm 6 are shown in Figure 16. Note that the overlay time mesh has the same number of time instances in total as in the pure state or pure adjoint adaptive case, such that results are comparable.
We consider the relative $L^2(0,T;\Omega)$-error between the analytical solution and the POD solution in the state and in the control variable given as

$$err_y = \frac{\|\bar{y} - y_\ell\|_{L^2(0,T;\Omega)}}{\|\bar{y}\|_{L^2(0,T;\Omega)}} \quad \text{and} \quad err_u = \frac{\|\bar{u} - u_\ell\|_{L^2(0,T;\Omega)}}{\|\bar{u}\|_{L^2(0,T;\Omega)}}$$

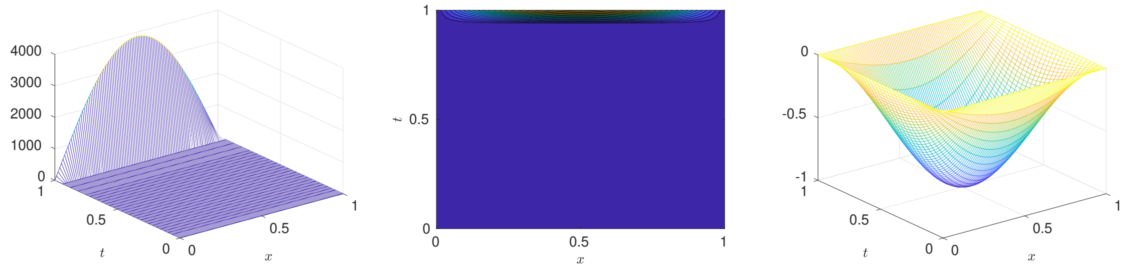

Figure 15: POD optimal state computed on the adaptive time grid with $n = 27$ (left), its contour lines (middle) and the POD optimal control using Algorithm 5 with input control $u = 0$ (right)

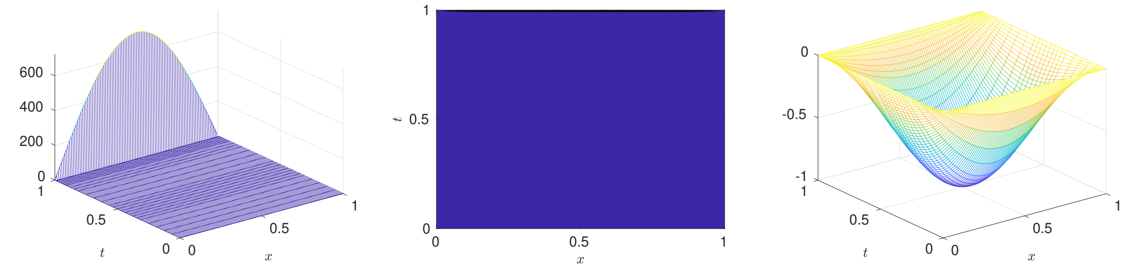

Figure 16: POD optimal state computed on the adaptive time grid with $n = 27$ (left), its contour lines (middle) and the POD optimal control using Algorithm 6 with input control $u = 0$ (right)

for different numbers of time instances n in Figure 17.

Concerning the error in the state variable, the advantage of using an adaptive time grid which is tailored for the state is obvious. For $n \leq 45$ it also delivers better results than the time grid according to Algorithm 6, since we compare the approximation results for the same number of time instances.

Regarding the error in the control variable, we observe that for all four approaches an enlargement of the number of time instances leads to a reduction in the error. The best results are achieved in this example if we consider the uniform or the adjoint adaptive method.

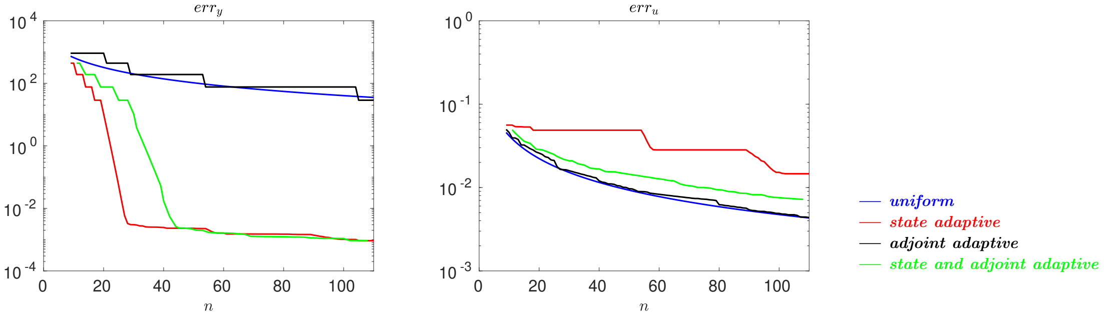

Figure 17: Relative $L^2(0, T; \Omega)$-errors between the analytical optimal solution and the POD optimal solution depending on n. Left: error in the state, right: error in the control

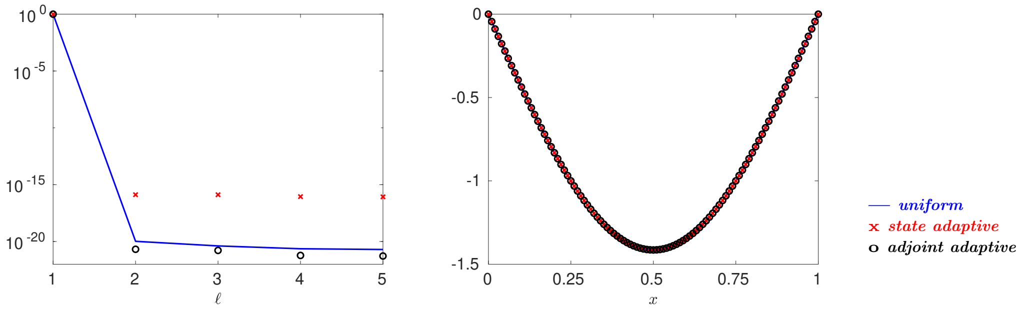

Figure 18: Decay of normalized eigenvalues (left) and first POD mode $\psi_1(x)$ using adaptive and uniform time grids, respectively (right)

Influence of POD basis length on the approximation quality

In Figure 18 (left) we see the decay of the eigenvalues for a uniform and an adaptive time discretization using $n = 27$. The POD basis is computed corresponding to the input control $u = 0$ from a joint state and adjoint state snapshot set. Moreover, Figure 18 (right) shows the first POD basis using a uniform mesh with $n = 27$ and the POD basis using an adaptive time grid with the same number of time points. We see that in this example, increasing the number of utilized POD basis functions will not lead to an improvement of the POD reduced-order accuracy since $\lambda_2 \leq 10^{-15}$, i.e. no more information will be added. Moreover, the first POD basis function computed with a uniform time discretization is numerically the same as the POD basis computed corresponding to a state or adjoint adaptive time discretization. The same accounts for the POD computations for the joint state and adjoint adaptive scheme, but is left out in the plots due to visibility. The reason why only one POD basis function is sufficient is due to the fact that the spatial structure of the optimal state and adjoint state is determined by the function $sin(\pi x)$. This choice is done by purpose since the focus of the numerical test lies on the snapshot location and we want to eliminate other dependencies. More POD basis functions will be needed if the spatial structure of the optimal state and/or the adjoint state cannot be determined by only one function.

Moreover, Figure 18 even tells us more: since the uniform time discretization and the adaptive time discretization lead to the exact same POD basis, the location of the snapshots does not influence the POD basis in this setting. The time locations play an important role in the optimization algorithm. We provide an example in Section 6.1, in which a dynamical signal is missed in case of a uniform snapshot location and the resulting snapshots are zero functions, such that the corresponding POD basis is an arbitrary orthonormal basis. In contrary, the adaptive snapshot location is able to detect the time signal and the corresponding POD basis captures the spatial structure satisfactorily.

Influence of the input control

We now investigate the influence of the input control. In Figure 19 (left) we see the space-time state solution y_{h^+} computed according to the state adaptive scheme of Algorithm 4 on a coarse spatial mesh with resolution $h^+ = 1/5$. Then, we compute an associated adjoint solution by solving (3.10) with right-hand side πy_{h^+}, where π denotes the Lagrange interpolation which interpolates the space-time discrete function $y_{h^+} \in Y_h$ for each time instance into the finite element space V_h. The corresponding control u_{h^+} is then computed according to (3.11) on a coarse spatial mesh which is shown in Figure 19 (middle). We interpolate this control onto a fine mesh with resolution $h^* = 1/100$, Figure 19 (right), and use this interpolation $\mathbb{I}u_{h^+}$ as input control for snapshot generation.

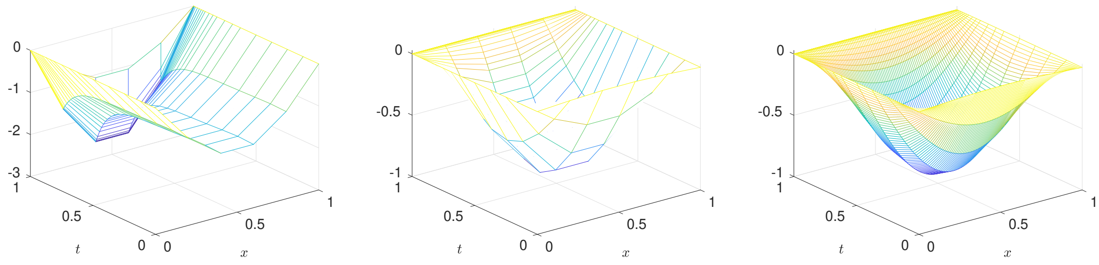

Figure 19: Space-time state y_{h^+} with $n = 27$ and $h^+ = 1/5$ (left), corresponding control u_{h^+} (middle) and interpolation $\mathbb{I}u_{h^+}$ of this control onto a fine mesh with $h^* = 1/100$ (right)

Next, we consider Algorithm 5. The adjoint space-time solution p_{h^+} with coarse spatial resolution $h^+ = 1/5$ as well as the with πp_{h^+} associated control u_{h^+} are shown in Figure 20 (left and middle). Finally, the interpolation of the control u_{h^+} according to Algorithm 6 onto a fine mesh is shown in Figure 20 (right). Note that this control constitutes a poor approximation of the optimal control since its appearance is dominated by the part stemming from the state adaptive scheme, but using only 19 time points (such that the joint time grid has 28 time points).

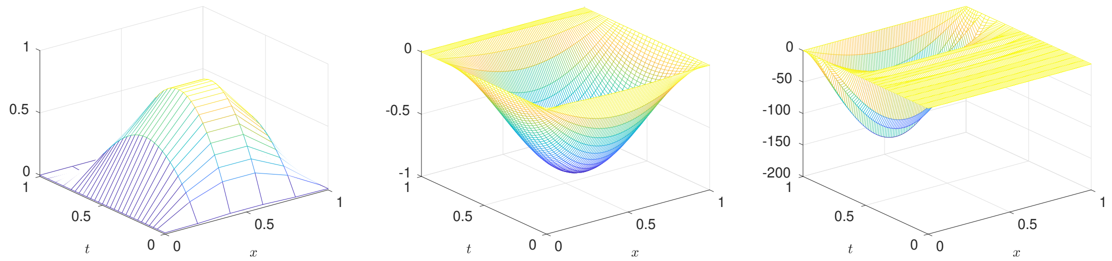

Figure 20: Space-time adjoint state p_{h^+} with $n = 27$ and $h^+ = 1/5$ (left), interpolation of corresponding control $\mathbb{I}u_{h^+}$ onto a fine mesh with $h^* = 1/100$ (middle) and interpolation of the control associated with the union of the space-time adaptive state and adjoint state (right)

We compare the influence of the input control for snapshot generation in Figure 21. It turns out, that for this problem setting, the input control has no influence on the approximation quality.

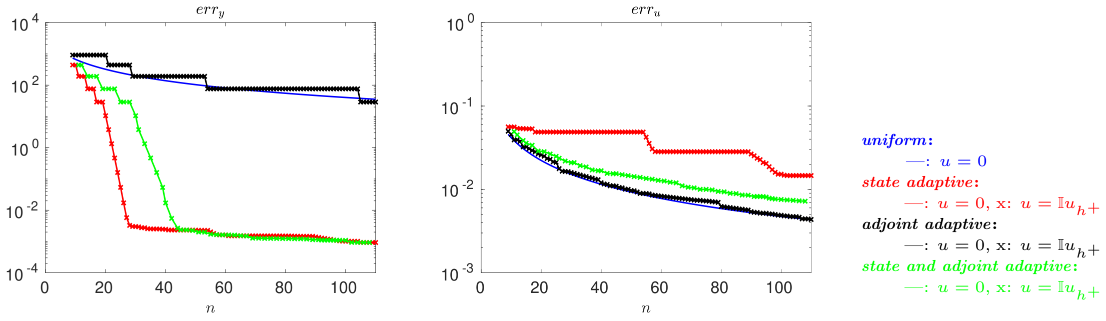

Figure 21: Relative $L^2(0, T; \Omega)$-errors between the analytical optimal solution and the reduced-order solution depending on n and comparing an input control $u = 0$ with an input control $u = \mathbb{I}u_{h^+}$ for snapshot generation. Left: error in the state, right: error in the control

Efficiency

We analyze the proposed methodology concerning computational efficiency versus accuracy in Table 1. If an accuracy of order 10^{-03} of the numerical solution in comparison to the true solution is aimed, then using the adaptive snapshot location tailored for the state (Algorithm 4) chooses $n = 27$ and computes the adaptive time grid in 0.61 seconds. The generation of snapshots is very fast with only 0.002 seconds. For the computation of the POD basis, we solve an eigenvalue problem for $\bar{Z}^T\bar{Z} = D^{1/2}ZWZD^{1/2}$, where Z denotes the matrix in which the state and adjoint snapshots are stored columnwise, $D = \mathrm{diag}(\alpha_0, \dots, \alpha_n)$ is a diagonal matrix containing the time weights and W is a matrix related to the inner product $(\cdot,\cdot)_X$ in (2.7), i.e. for this setting with $X = L^2(\Omega)$ it is the mass matrix. This eigenvalue problem for $\bar{Z}^T\bar{Z}$ is the discrete counterpart of a snapshot gramian which will be discussed in Section 4.2.2 in detail. Note that the dimension of $\bar{Z}^T\bar{Z}$ only depends on the number of time instances $n + 1$, such that the solution of the related eigenvalue problem is very cheap with only 0.09 seconds. Then, the online time for the solution of the POD reduced optimal control problem is only 0.01 seconds. Thus, the major time consumption lies in the construction of the time-adaptive grid. However, this effort pays off if we consider a comparison using a uniform time discretization. In order to achieve the same accuracy as in the adaptive reduced-order computation, we need in a full finite element solution $n = 30150$ time instances, which leads to a computational time of 10.30 seconds. So, the reduced-order model with adaptive snapshot location and including offline times leads to a speed up of factor 14.5.
In comparison, using a uniform snapshot location for POD reduced-order modeling leads to the following computational issues. Utilizing the snapshot gramian $\bar{Z}^T\bar{Z}$ for computing the POD basis reaches its limits at $n = 17500$ and results in an accuracy of $err_y = 1.50 \cdot 10^{-02}$, where the solution of the eigenvalue problem takes 70.18 seconds. For larger numbers of time instances, the solution of the eigenvalue problem is not possible on our computing machines due to memory problems. As an alternative, we can compute a POD basis from a singular value decomposition of the snapshot matrix which reaches its memory limits for $n = 22500$ leading to an accuracy of $err_y = 9.71 \cdot 10^{-03}$, where the time to compute the POD basis is 11.14 seconds. Another option is to solve an eigenvalue problem for $\bar{Z}\bar{Z}^T = W^{1/2}ZDZW^{1/2}$ which is the discrete analogon to the operator $\mathcal{R}$ in (2.10). The solution of an associated eigenvalue problem with $n = 30150$ takes 0.21 seconds and leads to a POD reduced-order accuracy of $5.77 \cdot 10^{-03}$. Note carefully that this method involves the computation of an inverse weighting matrix $W^{1/2}$ which is feasible in this setting due to the discretization level $h^* = 1/100$. However, if we consider a finer spatial resolution or a 2d or 3d setting, the inversion of a the matrix W might become intensive.
Moreover, we observe that the use of the adaptive snapshot grid for the solution of the reduced-order model in the online phase brings a speed up in comparison to the full-order computation (factor 1030) and the reduced-order solution (factor 51) in case of a uniform time grid.

3.7 Located control and control constraints

In many applications it is not possible to control the system everywhere in the spatial domain. For this reason, the control is often considered as a variable which only depends on time and is matched to the time-space domain via a control operator $\mathcal{B}$. We refer to this as located control. Let us consider the space $U = L^2(0, T; \mathbb{R}^m), m \in \mathbb{N}$, as control space. We introduce an admissible set of controls by

$$U_{ad} = \{u \in L^2(0, T; \mathbb{R}^m) : u_a(t) \leq u(t) \leq u_b(t) \text{ in } \mathbb{R}^m \text{ a.e. in } [0, T]\}, \tag{3.53}$$

with box constraints given by the lower and upper bounds $u_a, u_b \in L^\infty(0, T; \mathbb{R}^m)$ with $u_a(t) \leq u_b(t)$ almost everywhere in $(0, T)$. Note that U_{ad} is a non-empty, bounded, convex and closed

	uniform, FE	uniform, ROM	uniform, ROM	adaptive, ROM
		solve eigs($\bar{Z}\bar{Z}^T$)	solve eigs($\bar{Z}^T\bar{Z}$)	solve eigs($\bar{Z}^T\bar{Z}$)
n	30150	30150	17500	27
err_y	$5.77 \cdot 10^{-03}$	$5.77 \cdot 10^{-03}$	$1.50 \cdot 10^{-02}$	$5.77 \cdot 10^{-03}$
optimal control solution	10.30 sec	0.51 sec	0.40 sec	0.01 sec
snapshot generation	–	0.75 sec	0.43 sec	0.002 sec
POD basis	–	0.21 sec	70.18 sec	0.09 sec
time-adaptive grid	–	–	–	0.61 sec

Table 1: Accuracy and computational times for the solution of the optimal control problem using a uniform versus an adaptive time discretization according to Algorithm 4 comparing a full finite element method with spatial discretization $h^* = 1/100$ with a POD reduced-order model using $\ell = 1$ POD basis function. The POD basis is computed solving a weighted eigenvalue problem (eigs) for $\bar{Z}\bar{Z}^T$ or $\bar{Z}^T\bar{Z}$

subset of $L^\infty(0,T;\mathbb{R}^m)$. We introduce a control operator as

$$\mathcal{B} : U \to L^2(0,T;H^{-1}(\Omega)), \quad u \mapsto \mathcal{B}u = \sum_{i=1}^m u_i \chi_i,$$

where $\chi_i \in H^{-1}(\Omega), 1 \leq i \leq m$, denote given shape functions. Thus, $\mathcal{B}$ is a linear and bounded operator. The dual operator $\mathcal{B}^* : L^2(0,T;H^{-1}(\Omega))^* \to U^*$ fulfills

$$\begin{aligned}
\langle \mathcal{B}u, v\rangle_{L^2(0,T;H^{-1}(\Omega)),L^2(0,T;H_0^1(\Omega))} &= \int_0^T \langle \mathcal{B}u(t), v(t)\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} dt \\
&= \sum_{i=1}^m \int_0^T u_i(t)\langle \chi_i, v(t)\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} dt \\
&= \sum_{i=1}^m \int_0^T u_i(t)(\mathcal{B}^* v(t))_i dt \\
&= (u, \mathcal{B}^* v)_{L^2(0,T;\mathbb{R}^m)} = \langle u, \mathcal{B}^* v\rangle_U
\end{aligned}$$

according to Definition 2.9. Thus, it holds $(\mathcal{B}^* v(t))_i = \langle \chi_i, v(t)\rangle_{H^{-1}(\Omega),H_0^1(\Omega)}$. Note that we identify $L^2(0,T;H^{-1}(\Omega))^*$ with $L^2(0,T;H_0^1(\Omega))$ and U^* with U. In the state equation (3.1) we replace the distributed control by the located control as follows

$$\begin{aligned}
y_t - \Delta y &= \mathcal{B}u && \text{in } (0,T] \times \Omega, && (3.54a)\\
y &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.54b)\\
y(0,\cdot) &= y_0 && \text{in } \Omega. && (3.54c)
\end{aligned}$$

Existence of a unique weak solution $y \in W(0,T;H_0^1(\Omega))$ to (3.54) and the regularity results follow according to the theory for (3.1). We introduce the weak solution operator associated with equation (3.54) by

$$\mathcal{S}_{ad} : U_{ad} \times L^2(\Omega) \to W(0,T;H_0^1(\Omega)), \quad (u, y_0) \mapsto y := \mathcal{S}_{ad}(u, y_0). \tag{3.55}$$

As cost functional we consider the quadratic objective

$$J_{ad}(y,u) = \frac{1}{2}\|y - y_d\|^2_{L^2(0,T;\Omega)} + \frac{\alpha}{2}\|u\|^2_{L^2(0,T;\mathbb{R}^m)}.$$

The optimal control problem reads as

$$\min_{(y,u)\in W(0,T;H_0^1(\Omega))\times U_{ad}} J_{ad}(y,u) \quad \text{s.t.} \quad y = \mathcal{S}_{ad}(u, y_0). \tag{3.56}$$

Due to the control constraints, the gradient equation (3.11) in the optimality system is replaced by the variational inequality

$$(\alpha u + \mathcal{B}^* p, v - u)_U \geq 0 \quad \text{for all } v \in U_{ad}. \tag{3.57}$$

The resulting first-order necessary (and by convexity sufficient) optimality system is given by the state equation (3.54) together with the adjoint equation

$$\begin{aligned} -p_t - \Delta p &= y - y_d && \text{in } [0,T) \times \Omega, && (3.58a)\\ p &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.58b)\\ p(T,\cdot) &= 0 && \text{in } \Omega, && (3.58c) \end{aligned}$$

and the variational inequality (3.57) which is equivalent to the projection formula

$$u(t) = \mathbb{P}_{[u_a(t),u_b(t)]}\left\{-\frac{1}{\alpha}(\mathcal{B}^* p)(t)\right\} \quad \text{for almost all } t \in [0,T], \tag{3.59}$$

where we define by

$$\mathbb{P}_{[u_a(t),u_b(t)]} : \mathbb{R}^m \to [u_a(t), u_b(t)]$$

for almost all $t \in [0,T]$ the componentwise orthogonal projection onto the admissible set of control vectors. The projection can be computed as

$$\mathbb{P}_{[u_a(t),u_b(t)]}\left\{-\frac{1}{\alpha}(B^* p)(t)\right\} = \max\left(u_a(t), \min\left(u_b(t), -\frac{1}{\alpha}\langle \chi_i, p(t)\rangle_{H^{-1}(\Omega),H_0^1(\Omega)}\right)\right),$$

which is meant componentwise for almost all $t \in [0,T]$. Moreover, we denote by

$$\mathbb{P}_{[u_a,u_b]}\{u\} := \mathbb{P}_{[u_a(t),u_b(t)]}\{u(t)\}, \quad \text{for almost all } t \in [0,T]$$

the pointwise projection of a function $u \in L^\infty(0,T;\mathbb{R}^m)$ onto the set of admissible controls $u_a, u_b \in L^\infty(0,T;\mathbb{R}^m)$.

Reformulation of the optimality system with respect to the adjoint state

We now can only reduce the optimality system to an elliptic system in p. The derivation of the biharmonic system is given later in the proof of Theorem 3.16. The resulting biharmonic system reads as

$$\begin{aligned} -p_{tt} + \Delta^2 p - \mathcal{B}\,\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p\right\} &= -(y_d)_t + \Delta y_d && \text{in } (0,T) \times \Omega, && (3.60a)\\ p &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.60b)\\ \Delta p &= y_d && \text{on } [0,T] \times \partial\Omega, && (3.60c)\\ p(T,\cdot) &= 0 && \text{in } \Omega, && (3.60d)\\ (p_t + \Delta p)(0,\cdot) &= y_d(0,\cdot) - y_0 && \text{in } \Omega. && (3.60e) \end{aligned}$$

Note that a reduction with respect to the state variable is not possible in the standard way since the projection operator $\mathbb{P}_{[u_a,u_b]}$ is a non-smooth operator, thus non-differentiable in the classical sense. One possible way to derive a biharmonic equation for the state variable is to consider a regularization of the projection, see [139]. This option is not further discussed here and we continue with the focus on the biharmonic equation with respect to the adjoint variable.

Lemma 3.14. *Let us denote* $N(p) := -\mathcal{B}\,\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p\right\}$. *It holds true*

$$\int_0^T \int_\Omega (N(p_1) - N(p_2))(p_1 - p_2)\,dx\,dt \geq 0 \quad \forall p_1, p_2.$$

Proof. Since the projection operator is monotone increasing, i.e.

$$u_1(t) \le u_2(t) \quad \Rightarrow \quad \mathbb{P}_{[u_a(t),u_b(t)]}\left\{u_1(t)\right\} \le \mathbb{P}_{[u_a(t),u_b(t)]}\left\{u_2(t)\right\}, \quad t \in [0,T]$$

for $u_1, u_2 \in L^\infty(0,T;\mathbb{R}^m)$ it is easy to see that

$$\left(\mathbb{P}_{[u_a(t),u_b(t)]}\{u_1(t)\} - \mathbb{P}_{[u_a(t),u_b(t)]}\{u_2(t)\}, u_1(t) - u_2(t)\right)_{\mathbb{R}^m} \ge 0$$

We set $u_1(t) := -\frac{1}{\alpha}(\mathcal{B}^* p_1)(t)$ and $u_2(t) := -\frac{1}{\alpha}(\mathcal{B}^* p_2)(t)$ for $p_1, p_2 \in \tilde{H}_0^{2,1}(0,T;\Omega)$. Then it holds

$$\left(-\mathbb{P}_{[u_a(t),u_b(t)]}\left\{-\frac{1}{\alpha}(\mathcal{B}^* p_1)(t)\right\} + \mathbb{P}_{[u_a(t),u_b(t)]}\left\{-\frac{1}{\alpha}(\mathcal{B}^* p_2)(t)\right\}, (\mathcal{B}^* p_1)(t) - (\mathcal{B}^* p_2)(t)\right)_{\mathbb{R}^m} \ge 0.$$

Hence by linearity of $\mathcal{B}^*$ it follows

$$\langle -\mathcal{B}\mathbb{P}_{[u_a(t),u_b(t)]}\left\{-\frac{1}{\alpha}(\mathcal{B}^* p_1)(t)\right\} + \mathcal{B}\mathbb{P}_{[u_a(t),u_b(t)]}\left\{-\frac{1}{\alpha}(\mathcal{B}^* p_2)(t)\right\}, p_1(t) - p_2(t)\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} \ge 0$$

which leads to the claim. □

Thus, the quantity

$$-\mathcal{B}\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p\right\}$$

is monotone in p and (3.60) is a semilinear elliptic problem with monotone nonlinearity.

In order to derive the weak formulation for (3.60) we define the operator A_0^{ad} as

$$A_0^{ad} : \tilde{H}_0^{2,1}(0,T;\Omega) \times \tilde{H}_0^{2,1}(0,T;\Omega) \to \mathbb{R},$$

$$A_0^{ad}(p,v) = \int_0^T \int_\Omega \left(p_t v_t - \mathcal{B}\,\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p\right\} v + \Delta p \Delta v\right) dxdt + \int_\Omega \nabla p(0)\nabla v(0)dx.$$

Then the weak form of equation (3.60) reads as

$$A_0^{ad}(p,v) = L_0(v) \quad \forall v \in \tilde{H}_0^{2,1}(0,T;\Omega), \tag{3.61}$$

where the linear form L_0 is defined in (3.22).

Theorem 3.15. (Existence of a unique weak solution)**.** *For every $L_0 \in (\tilde{H}_0^{2,1}(0,T;\Omega))^*$ the equation* (3.61) *has a unique solution $p \in \tilde{H}_0^{2,1}(0,T;\Omega)$.*

Proof. In analogy to [139, Lemma 4.6] it can be shown that the operator A_0^{ad} is strongly monotone, coercive and hemi-continuous. Then, the main theorem on monotone operators (Theorem 2.25) ensures existence of a unique solution. □

Equivalence of the optimality system to a biharmonic equation

Theorem 3.16. *Let $(y,u) \in W(0,T;H_0^1(\Omega)) \times U_{ad}$ denote the solution to problem* (3.56) *with associated adjoint state $p \in W(0,T;H_0^1(\Omega))$. Assume that $y_0 \in H_0^1(\Omega)$ and $y_d \in L^2(0,T;H^2(\Omega)) \cap H^1(0,T;L^2(\Omega))$. Further we assume the compatibility condition $y(T) - y_d(T) + \Delta p(T) \in H_0^1(\Omega)$ to hold true. Then p satisfies* (3.60) *a.e. in space-time and is a weak solution to* (3.60).

Proof. Due to $y_0 \in H_0^1(\Omega)$ we follow by Remark 3.2 that $y \in L^2(0,T;H^2(\Omega)) \cap H^1(0,T;L^2(\Omega))$. Then, we can apply Remark 3.3 which leads to the regularity

$$p \in L^2(0,T;H^4(\Omega)) \cap H^1(0,T;H^2(\Omega)) \cap H^2(0,T;L^2(\Omega)).$$

Thus, we can take in (3.58a) the derivative with respect to time in order to get

$$-p_{tt} - \Delta p_t = y_t - (y_d)_t.$$

Then, we replace y_t using (3.54a) and obtain

$$-p_{tt} - \Delta p_t = \Delta y + \mathcal{B}u - (y_d)_t.$$

We use the projection formula (3.59) in order to eliminate the control variable and get

$$-p_{tt} - \Delta p_t = \Delta y + \mathcal{B}\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p\right\} - (y_d)_t.$$

Finally, we use the identity $y = -p_t - \Delta p + y_d$ from (3.58a) in order to eliminate the state variable and get

$$-p_{tt} - \Delta p_t = \Delta(-p_t - \Delta p + y_d) + \mathcal{B}\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p\right\} - (y_d)_t,$$

which implies (3.60a). We get the initial condition (3.60e) by evaluating (3.58a) at $t = 0$. The boundary condition (3.60c) is derived by evaluating (3.58a) on the boundary $\partial\Omega$.
Thus, the adjoint function $p \in L^2(0,T;H^4(\Omega)) \cap H^1(0,T;H^2(\Omega)) \cap H^2(0,T;L^2(\Omega))$ fulfills the biharmonic problem (3.60) a.e. in space-time. Then, $p \in \tilde{H}_0^{2,1}(0,T;\Omega)$ is a weak solution to (3.60). □

Since the biharmonic equation (3.60) has a unique weak solution according to Theorem 3.15 and the weak solution p to (3.58) is a weak solution to (3.60) according to Theorem 3.16, the other direction holds, i.e. the weak solution to (3.60) is the optimal adjoint state p.

A-posteriori error estimate for the time discretization

We consider the same notation as in Section 3.4. The time-discrete problem for (3.61) reads as follows: find $p_k \in \tilde{\mathscr{V}}_k$ with $p_k(T,\cdot) = 0$ and

$$A_0^{ad}(p_k, v_k) = L_0(v_k) \quad \forall v_k \in \tilde{\mathscr{V}}_k. \tag{3.62}$$

The time-discrete problem has a unique solution by construction. We derive and prove a temporal residual-type a-posteriori error estimate for the variable p in the spirit of [82]. For this, we need the following results and properties.

Lemma 3.17. *For $\alpha > 0$ and $v \in \tilde{H}_0^{2,1}(0,T;\Omega)$ it holds*

$$\|v\|_{H^{2,1}(0,T;\Omega)} \leq c\left(\|v_t\|^2_{L^2(0,T;\Omega)} + \|\Delta v\|^2_{L^2(0,T;\Omega)}\right)$$

with constant $c > 0$.

Proof. We define the function $g := -v_t - \Delta v \in L^2(0,T;\Omega)$. Then, v is the solution to

$$\begin{aligned} -v_t - \Delta v &= g \quad \text{in } [0,T) \times \Omega, \\ v &= 0 \quad \text{on } [0,T] \times \partial\Omega, \\ v(T,\cdot) &= 0 \quad \text{in } \Omega, \end{aligned}$$

in a weak sense. This solution v depends continuously on the data (compare e.g. (3.14)), thus

$$\begin{aligned}\|v\|^2_{H^{2,1}(0,T;\Omega)} &\leq c\|g\|^2_{L^2(0,T;\Omega)}\\ &= c\|-v_t-\Delta v\|^2_{L^2(0,T;\Omega)}\\ &\leq 2c\left(\|v_t\|^2_{L^2(0,T;\Omega)}+\|\Delta v\|^2_{L^2(0,T;\Omega)}\right),\end{aligned}$$

where we applied Young's inequality, Theorem 2.31.

□

Lemma 3.18. *It holds*

$$A_0^{ad}(p,v_k)-A_0^{ad}(p_k,v_k)=0 \quad \forall v_k\in\tilde{\mathscr{V}}_k.$$

Proof. With (3.61) and (3.62) we have

$$A_0^{ad}(p,v_k)-A_0^{ad}(p_k,v_k)=L_0(v_k)-L_0(v_k)=0 \quad \forall v_k\in\tilde{\mathscr{V}}_k\subset\tilde{H}_0^{2,1}(0,T;\Omega).$$

□

The following theorem states a temporal residual-type a-posteriori estimate for p which transfers the result in Theorem 3.7 to the case of located control and control constrained optimal control.

Theorem 3.19. *Let $p\in\tilde{H}_0^{2,1}(0,T;\Omega)$ and $p_k\in\tilde{\mathscr{V}}_k$ denote the solutions to (3.61) and (3.62), respectively. Then, we obtain the a-posteriori error estimation*

$$\|p-p_k\|^2_{H^{2,1}(0,T;\Omega)}\leq c\,\eta^2_{ad}, \tag{3.63}$$

where $c>0$ and

$$\eta^2_{ad}=\sum_{j=1}^n\Delta t_j^2\int_{I_j}\left\|-(y_d)_t+\Delta y_d+(p_k)_{tt}+\mathcal{B}\,\mathbb{P}_{U_{ad}}\left\{-\frac{1}{\alpha}\mathcal{B}^*p_k\right\}-\Delta^2p_k\right\|^2_{L^2(\Omega)}$$
$$+\sum_{j=1}^n\int_{I_j}\|y_d-\Delta p_k\|^2_{L^2(\partial\Omega)}.$$

Proof. We set $e^p:=p-p_k, N(p):=-\mathcal{B}\,\mathbb{P}_{U_{ad}}\left\{-\frac{1}{\alpha}\mathcal{B}^*p\right\}$ and denote by $\pi e^p\in\tilde{\mathscr{V}}_k$ the Lagrange interpolation of e^p. We make the following estimations

$$\begin{aligned}\frac{1}{c}\|p-p_k\|^2_{H^{2,1}(0,T;\Omega)} &\leq \|(p-p_k)_t\|^2_{L^2(0,T;\Omega)}+\|\Delta(p-p_k)\|^2_{L^2(0,T;\Omega)} && \text{(Lemma 3.17)}\\ &\leq \int_0^T\int_\Omega(p-p_k)_te_t^p+\Delta(p-p_k)\Delta e^p+(N(p)-N(p_k))e^pdxdt\\ &&& \text{(Lemma 3.14)}\\ &\leq \int_0^T\int_\Omega(p-p_k)_te_t^p+\Delta(p-p_k)\Delta e^p+(N(p)-N(p_k))e^pdxdt\\ &\quad +\|\nabla(p-p_k)(0)\|^2_{L^2(\Omega)}\\ &= A_0^{ad}(p,e^p)-A_0^{ad}(p_k,e^p)\\ &= A_0^{ad}(p,e^p)-A_0^{ad}(p_k,e^p)-(A_0^{ad}(p,\pi e^p)-A_0^{ad}(p_k,\pi e^p)) && \text{(Lemma 3.18)}\\ &= A_0^{ad}(p,e^p-\pi e^p)-A_0^{ad}(p_k,e^p-\pi e^p)\\ &= L_0(e^p-\pi e^p)-A_0^{ad}(p_k,e^p-\pi e^p)\end{aligned}$$

$$\begin{aligned}
&= \int_0^T \int_\Omega (-(y_d)_t + \Delta y_d)(e^p - \pi e^p)dxdt - \int_\Omega (y_d(0) - y_0)(e^p(0) - (\pi e^p)(0))dx \\
&\quad + \int_0^T \int_{\partial\Omega} y_d \nabla(e^p - \pi e^p) \cdot \vec{n}\, dsdt - \int_\Omega \nabla p_k(0)\nabla(e^p - \pi e^p)(0)dx \\
&\quad - \int_0^T \int_\Omega ((p_k)_t(e^p - \pi e^p)_t + N(p_k)(e^p - \pi e^p) + \Delta p_k \Delta(e^p - \pi e^p))dxdt \\
&= \int_0^T \int_\Omega (-(y_d)_t + \Delta y_d + (p_k)_{tt} - N(p_k) - \Delta^2 p_k)(e^p - \pi e^p)dxdt \\
&\quad + \int_0^T \int_{\partial\Omega} (y_d - \Delta p_k)\nabla(e^p - \pi e^p) \cdot \vec{n}\, dsdt \qquad \text{(integration by parts and} \\
&\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{Theorem 2.20).}
\end{aligned}$$

Splitting the temporal integration into an integration over each time interval, using error estimates of the Lagrange interpolation π, the trace inequality and Young's inequality (Theorem 2.31), we get the claim.

□

Space-time mixed finite element discretization

We state the space-time mixed finite element variational form for (3.60) which is derived along the lines to (3.33) and uses the same notations. It reads as: find $p_h \in \tilde{Y}_0^h, \vartheta_h \in \tilde{\mathscr{V}}_k$ and $\vartheta_h|_{\partial\Omega} = -\tilde{y}_d$ such that

$$\begin{aligned}
\int_0^T \int_\Omega (p_h)_t v_t - \mathcal{B}\mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p_h\right\} v + \nabla\vartheta_h \nabla v dxdt + \int_\Omega \nabla p_h(0)\nabla v(0)dx& \\
= \int_0^T \int_\Omega (-(y_d)_t + \Delta y_d)v\, dxdt - \int_\Omega (y_d(0) - y_0)v(0)dx \quad \forall v \in \tilde{Y}_0^h,&
\end{aligned} \tag{3.64a}$$

$$\int_0^T \int_\Omega -\nabla p_h \nabla\phi + \vartheta_h \phi\, dxdt = 0 \quad \forall \phi \in W_0^h. \tag{3.64b}$$

The adaptive finite element cycle which utilizes the error estimate (3.63) in order to adapt the time discretization according to the temporal dynamics of the adjoint state follows along the lines of Section 3.5 and is summarized in Algorithm 8.

Algorithm 8 Space-time finite element algorithm with time adaptivity for the adjoint state in case of control constraints and located control.

Input: Initial space-time mesh $\mathscr{T}_h^{init}$, tolerance $\varepsilon > 0$, refinement parameter $\theta \in (0,1)$, desired state y_d, initial state y_0, regularization parameter $\alpha > 0$, initial number of time discretization points n, maximal number of time points $n_{\max}$.

Output: Time-adapted space-time mesh $\mathscr{T}_h$, adjoint solution p_h.

```
1: $\mathscr{T}_h = \mathscr{T}_h^{init}$ with $n$ time points
2: while $n < n_{\max}$ OR $\eta_p \geq \varepsilon$ do
3:     Define $\tilde{\mathscr{V}}_k$ as the finite element space corresponding to $\mathscr{T}_h$ according to (3.30).
4:     Solve (3.64).
5:     Estimate the error contributions from (3.63).
6:     Mark time intervals according to the Dörfler criterion (Definition 2.34) with parameter $\theta$.
7:     Refine the marked time intervals using bisection.
8: end while
```

Remark 3.20. *Note that* (3.64) *is a nonlinear non-smooth equation due to the projection such that the solution can be done by a semi-smooth Newton method or a Newton method utilizing a regularization of the projection formula, for example. In the numerical tests, we will use a fixed point iteration in order to find an approximate solution.*

3.8 Snapshot location in optimal control for POD model order reduction with located control and control constraints

3.8.1 Methodology

The methodology of selecting appropriate time instances related to the optimal adjoint works analogously to the approach (ii) in Section 3.6 tailored for the adjoint state with the difference that instead of the biharmonic system (3.21) with error estimate (3.27) we now consider the equation (3.60) with error estimate (3.63). The approach is summarized in the following Algorithm 9.

Algorithm 9 Adaptive snapshot selection for linear optimal control problems with located control and control constraints.

```
Input: Coarse uniform spatial grid with resolution $h^+$.
Output: Snapshot locations $\{t_j\}_{j=0}^n$, approximation of the optimal control $u_{h^+}$.
1: Call Algorithm 8 in order to solve (3.60) adaptively w.r.t. time using the error estimator
   (3.63) with spatial resolution $h^+$. Obtain time instances $\{t_j\}_{j=0}^n = \mathcal{I}$ and the solution $p_{h^+}$.
2: Set $u_{h^+} = \mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}(\mathcal{B}^* p_{h^+})\right\}$ according to (3.59).
```

After the selection of suitable time instances for snapshot generation which are related to the optimal adjoint state and thus related to the optimal control, we proceed with the usual POD offline phase, in which state and adjoint state snapshots are sampled and a POD basis is computed. Finally, the POD online phase is executed in order to compute a POD reduced-order solution to the optimal control problem (3.56), which reads as

$$\min_{(y_\ell,u_\ell)\in W(0,T;X_\ell)\times U_{ad}} J_{ad}^\ell(y_\ell,u_\ell) \quad \text{s.t.} \quad y_\ell = \mathcal{S}_{ad}^\ell(u_\ell,\tilde{y}_0), \tag{3.65}$$

where the linear operator

$$\mathcal{S}_{ad}^\ell : U_{ad}\times X_\ell \to W(0,T;X_\ell), \quad (\hat{u},\tilde{y}_0)\mapsto y_\ell := \mathcal{S}_\ell(\hat{u},\tilde{y}_0)$$

is defined analogously to the solution operator $\mathcal{S}_{ad}$ defined in (3.55) associated with the reduced state equation given by

$$\frac{d}{dt}(y_\ell(t),\psi)_{L^2(\Omega)} + (\nabla y_\ell(t),\nabla\psi)_{L^2(\Omega)} = (\mathcal{B}u_\ell(t),\psi)_{L^2(\Omega)} \quad \text{for } t\in[0,T], \psi\in X_\ell. \tag{3.66}$$

The procedure is summed up in Algorithm 10.

Algorithm 10 POD model order reduction for linear-quadratic optimal control problems with control constraints and located control using adaptive snapshot locations.

```
Input: Snapshot locations $\{t_j\}_{j=0}^n = \mathcal{I}$ and input control $u_{h^+}$ from Algorithm 9, fine spatial
   resolution $h^*$, number of POD basis functions $\ell$
Output: $y_\ell, u_\ell$
1: Sample state and adjoint snapshots in a simulation of (3.54) and (3.58) with spatial resolution
   $h^*$ and input control $\mathbb{I}u_{h^+}$ on $\mathcal{I}$.
2: Compute a POD basis of rank $\ell$ according to (2.7) using an integrated snapshot ensemble.
3: Set up and solve the POD-ROM (3.65) on $\mathcal{I}$ to obtain $y_\ell, u_\ell$.
```

3.8.2 Error analysis

Analogously to the error analysis in Section 3.6.2, we derive error estimates for the reduced-order solution computed with Algorithm 10. The notation follows along the lines. In particular, we denote by $(\bar{y}, \bar{u})$ the solution to (3.56) with associated optimal adjoint state $\bar{p}$. The time-discrete solution to (3.62) is denoted by p_k and we define $u_k := \mathbb{P}_{[u_a,u_b]}\left\{-\frac{1}{\alpha}\mathcal{B}^* p_k\right\}$.

Error bound for the adjoint state

Let us start with estimating the error between the optimal adjoint state and the time-discrete POD approximation. Using the null-trick and applying the triangle inequality we arrive at

$$\|\bar{p} - \bar{p}_k^\ell\| \leq \underbrace{\|\bar{p} - p_k\|}_{(3.67.i)} + \underbrace{\|p_k - \tilde{p}_k(u_k)\|}_{(3.67.ii)} + \underbrace{\|\tilde{p}_k(u_k) - \tilde{p}_k^\ell(u_k)\|}_{(3.67.iii)} + \underbrace{\|\tilde{p}_k^\ell(u_k) - \bar{p}_k^\ell\|}_{(3.67.iv)}. \tag{3.67}$$

The first term (3.67.i) can be estimated utilizing the a-posteriori error estimate (3.63) for the adjoint state and concerns the snapshot location. It holds true

$$\|\bar{p} - p_k\|^2 \leq c_1 \eta_{ad}^2. \tag{3.68}$$

The estimation of the following two terms (3.67.ii) and (3.67.iii) are in line with the estimation for (3.46.ii) and (3.46.iii). Finally, the last term (3.67.iv) can be estimated in terms of the control variable using the a-posteriori error estimation from [177]. In particular, it holds

$$\|\tilde{p}_k^\ell(u_k) - \bar{p}_k^\ell(\bar{u}_k^\ell)\| \leq c\|u_k - \bar{u}_k^\ell\|_{L^2(0,T;\mathbb{R}^m)} \leq c\|u_k - \bar{u}\|_{L^2(0,T;\mathbb{R}^m)} + c\|\bar{u} - \bar{u}_k^\ell\|_{L^2(0,T;\mathbb{R}^m)}, \tag{3.69}$$

with constant $c > 0$ from (3.14). The error between the (unknown) optimal control $\bar{u}$ and a particular control u_p (here $u_p = u_k$ and $u_p = \bar{u}_k^\ell$, respectively) can be estimated via

$$\|\bar{u} - u_p\|_{L^2(0,T;\mathbb{R}^m)} \leq \frac{1}{\alpha}\|\zeta_p\|_{L^2(0,T;\mathbb{R}^m)}, \tag{3.70}$$

where the perturbation $\zeta_p \in L^2(0,T;\mathbb{R}^m)$ is chosen such that

$$(\alpha u_p + \mathcal{B}^* p_p + \zeta_p, v - u_p)_{L^2(0,T;\mathbb{R}^m)} \geq 0 \quad \text{for all } v \in U_{ad}$$

holds true where p_p denotes the adjoint state associated with u_p.
In conclusion, we establish the following theorem.

Theorem 3.21. *Let $\bar{p}$ denote the optimal adjoint state associated with* (3.56) *and let $\bar{p}_k^\ell$ denote the time-discrete adjoint solution associated with* (3.65)*. The POD basis is computed from an integrated snapshot set with input control and snapshot locations computed according to the Algorithm 9. In* (2.5) *it is either $X = V$ or $X = H$. Then, it holds true*

$$\begin{aligned}\|\bar{p} - \bar{p}_k^\ell\|_{L^2(0,T;X)} \leq \;& \sqrt{c_1}\eta_{ad} + \varepsilon + \left(c_2 \sum_{i>\ell} \lambda_i\right)^{1/2} + c_2\|(\tilde{p}_k(u_k))_t - \mathcal{P}^\ell(\tilde{p}_k(u_k))_t\|^2_{L^2(0,T;V^*)} \\ & + c_2\|\tilde{y}_k(u_k) - \tilde{y}_k^\ell(u_k)\|^2_{L^2(0,T;H)} + \frac{c_3}{\alpha}(\|\zeta_1\|_{L^2(0,T;\mathbb{R}^m)} + \|\zeta_2\|_{L^2(0,T;\mathbb{R}^m)}),\end{aligned} \tag{3.71}$$

with constants $c_1, c_2, c_3 > 0$ and $\zeta_1, \zeta_2 \in L^2(0,T;\mathbb{R}^m)$ are chosen such that

$$(\alpha u_k + \mathcal{B}^*\tilde{p}_k(u_k) + \zeta_1, v - u_k)_{L^2(0,T;\mathbb{R}^m)} \geq 0 \text{ and } (\alpha\bar{u}_k^\ell + \mathcal{B}^*\tilde{p}_k(\bar{u}_k^\ell) + \zeta_2, v - \bar{u}_k^\ell)_{L^2(0,T;\mathbb{R}^m)} \geq 0$$

for all $v \in U_{ad}$, respectively, is fulfilled.

Proof. Utilizing the decomposition (3.67) together with (3.68), (3.38b), (3.48), (3.49) and (3.70) lead to the claim. □

Remark 3.22. *Note that in the error estimate* (3.71) *in Theorem 3.21, there exists a dependency on the time derivatives, for which we refer to Remark 3.11(i). Moreover, the dependency on the state* $\|\tilde{y}_k(u_k) - \tilde{y}_k^\ell(u_k)\|^2_{L^2(0,T;H)}$ *can be estimated analogously to the term (3.39.iii). This concerns the estimation between the state snapshots and the POD state associated with the control* u_k*, which is the input control in order to generate the snapshots.*

Error bound for the control

The error regarding the control can be estimated using the a-posteriori error estimate from [177]. Moreover, we can estimate the error in the control variable with regard to the error in the adjoint variable. In particular, due to the optimality condition (3.57) it holds

$$(\alpha\bar{u} + \mathcal{B}^*\bar{p}, v - \bar{u})_{L^2(0,T;\mathbb{R}^m)} \geq 0 \quad \forall v \in U_{ad}, \tag{3.72a}$$

$$(\alpha\bar{u}_k^\ell + \mathcal{B}^*\bar{p}_k^\ell, v - \bar{u}_k^\ell)_{L^2(0,T;\mathbb{R}^m)} \geq 0 \quad \forall v \in U_{ad}. \tag{3.72b}$$

Taking $v = \bar{u}_k^\ell$ as admissible test function in (3.72a) and $v = \bar{u}$ as admissible test function in (3.72b) and adding both inequalities leads to

$$(\alpha(\bar{u} - \bar{u}_k^\ell) + \mathcal{B}^*(\bar{p} - \bar{p}_k^\ell), \bar{u}_k^\ell - \bar{u})_{L^2(0,T;\mathbb{R}^m)} \geq 0.$$

Thus, we can infer the estimation

$$\|\bar{u} - \bar{u}_k^\ell\|_{L^2(0,T;\mathbb{R}^m)} \leq \frac{1}{\alpha}\|\mathcal{B}^*(\bar{p} - \bar{p}_k^\ell)\|_{L^2(0,T;\mathbb{R}^m)}.$$

Error bound for the state

Regarding the error in the state variable $\|\bar{y} - \bar{y}_k^\ell\|$ we cannot guarantee that the time grid constructed via Algorithm 9 is a suitable time grid for the state variable. Since it is not possible to reformulate the optimality system into a biharmonic equation depending on the state and derive corresponding error estimates, we introduce a post-processing step. In this post-processing step, we undertake a further refinement of the adaptive time grid from Algorithm 8 in order to make it suitable for the state. If we take $X = V$ in (2.7), then in this situation, the following estimate is proved in [121]:

$$\sum_{j=0}^{n} \alpha_j \|y(t_j; u_k^\ell) - y_j^\ell(u_k^\ell)\|_H^2 \leq \sum_{j=1}^{n} \Big(\Delta t_j^2 C_y((1 + c_p^2)\|y_{tt}(u_k^\ell)\|^2_{L^2(I_j,H)} + \|y_t(u_k^\ell)\|_{L^2(I_j;V)})\Big) \tag{3.73a}$$

$$+ \sum_{j=1}^{n} C_y \left(\sum_{i=\ell+1}^{d} (|\langle\psi_i, y_0\rangle_V|^2 + \lambda_i) \right) \tag{3.73b}$$

$$+ \sum_{j=1}^{n} \sum_{i=\ell+1}^{d} C_y \frac{\lambda_i}{\Delta t_j^2} \tag{3.73c}$$

where $C_y > 0$ is a constant depending on T, but independent of the time grid $\{t_j\}_{j=0}^n$. We note that $y(t_j; u_k^\ell)$ is the continuous solution of (3.54) at given time instances related to the suboptimal control u_k^ℓ. The constant c_p is an upper bound of the projection operator. A similar estimate can be carried out if $X = H$. Estimate (3.73) provides now a recipe for further refinement of the time grid in order to approximate the state y within a prescribed tolerance. One option here consists in equidistributing the error contributions of the term (3.73a), while the number of modes has to be adapted to the time grid size according to the term (3.73c). Finally, the number ℓ of modes should be chosen such that the term in (3.73b) remains within the prescribed tolerance.

3.8.3 Numerical results

In order to validate the snapshot location strategy, we construct a numerical example such that we know the analytical solution. Similar examples are presented in [11]. All coding is done in MATLAB R2018a. The solution of the nonlinear, non-smooth biharmonic equation can be done by a semi-smooth Newton method or by a regularization of the projection formula (Remark 3.20). In this numerical test we compute an approximate solution to the biharmonic system with a fixed point iteration and initialize the method with the adjoint state corresponding to the unconstrained optimal control problem. In this way, only two iterations are needed for convergence. Convergence of the fixed point iteration can be argued for large enough values of α.

Problem setting

The spatial domain is chosen to be $\Omega = (0, 1) \subset \mathbb{R}$. We fix the end time $T = 1$ and choose the regularization parameter $\alpha = 1$. As in Section 3.6.3, we introduce a forcing term f on the right-hand side of the state equation (3.54) leading to a right-hand side f in (3.60a).

As optimal state we choose

$$\bar{y}(t, x) = x(x - 1)\sin(\pi t)$$

and thus, the initial condition is $y_0(x) = 0$. The optimal adjoint state is set to

$$\bar{p}(t, x) = \sin(\pi x)\left(t - \frac{e^{(t-1)/\varepsilon} - e^{-1/\varepsilon}}{1 - e^{-1/\varepsilon}}\right).$$

For small values of ε (we choose $\varepsilon = 10^{-03}$), the adjoint state $\bar{p}$ develops a layer towards $t = 1$, which can be seen Figure 23. We choose $m = 2$ control shape functions given by

$$\chi_1(x) = \max\{0, 1 - 16(x - 0.25)^2\}, \quad \chi_2(x) = \max\{0, 1 - 16(x - 0.75)^2\},$$

which are shown in Figure 24. This leads to

$$(\mathcal{B}^*\bar{p}(t))_1 = \int_\Omega \chi_1 \bar{p}(t)dx = (\mathcal{B}^*\bar{p}(t))_2 = \int_\Omega \chi_2 \bar{p}(t)dx = \left(\frac{32}{\pi^3} - \frac{8}{\pi^2}\right)\left(t - \frac{e^{(t-1)/\varepsilon} - e^{-1/\varepsilon}}{1 - e^{-1/\varepsilon}}\right).$$

We impose control constraints from below and above as $u_a(t) = -0.1$ and $u_b(t) = +0.1$. The analytical optimal controls are given by

$$\bar{u}_1(t) = \bar{u}_2(t) = \mathbb{P}_{[-0.1,0.1]}\left\{-\frac{1}{\alpha}\left(\frac{32}{\pi^3} - \frac{8}{\pi^2}\right)\left(t - \frac{e^{(t-1)/\varepsilon} - e^{-1/\varepsilon}}{1 - e^{-1/\varepsilon}}\right)\right\}$$

and are shown in Figure 24. The resulting source term f and the desired state y_d are constructed accordingly.

Uniform vs. adaptive snapshot location

In contrast to Algorithm 10, we choose as input controls $u_1(t) = u_2(t) = 0$ in order to create a fair comparison between a uniform and adaptive snapshot location strategy which only depends on the time discretization. We select $\ell = 1$ POD mode and use the same time grid for snapshot location and optimization.

For $n = 40$, the adaptive time grid computed according to Algorithm 9 as well as the associated adjoint state and controls are shown in Figure 25.

We illustrate the POD optimal adjoint state, its contour lines and the POD state computed on a uniform time mesh and on an adaptive time grid in Figures 26 and 27, respectively. The control

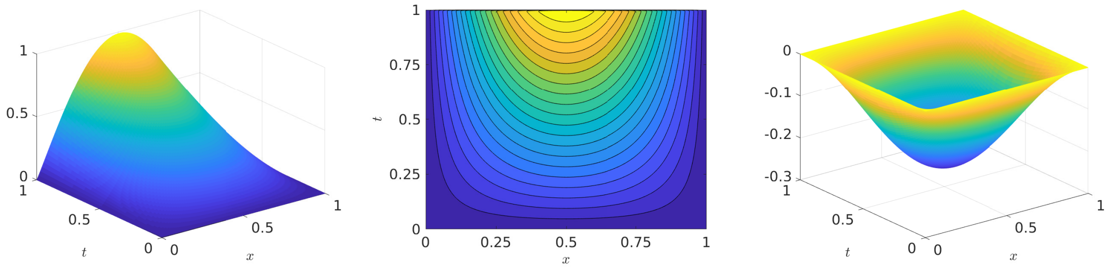

Figure 22: Analytical adjoint state $\bar{p}$ (left), its contour lines (middle) and analytical state $\bar{y}$ (right)

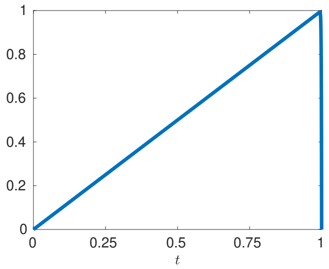

Figure 23: Time layer function $t - \frac{e^{(t-1)/\varepsilon} - e^{-1/\varepsilon}}{1-e^{-1/\varepsilon}}$

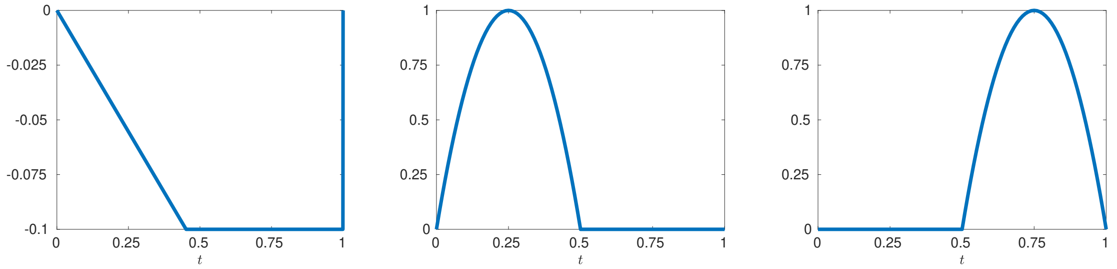

Figure 24: Analytical controls $\bar{u}_1 = \bar{u}_2$ (left), control shape functions χ_1 (middle) and χ_2 (right)

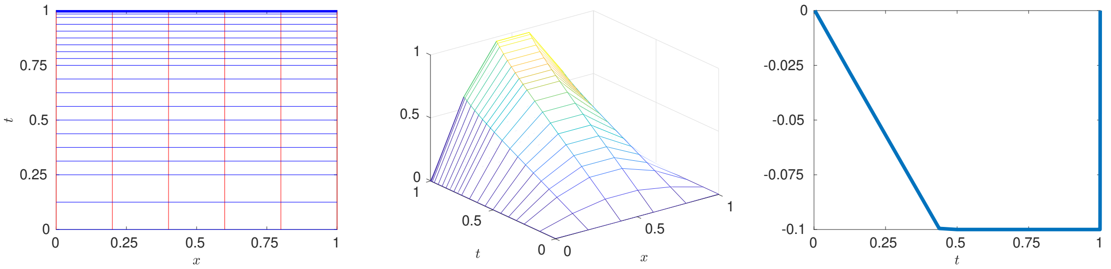

Figure 25: Adaptive space-time grid according to Algorithm 9 with $n = 40$ and $h^+ = 1/5$ (left), space-time adjoint state p_{h^+} (middle) and corresponding control $(u_{h^+})_1 = (u_{h^+})_2$ (right)

functions are shown in Figure 28. We observe that the use of the uniform time mesh leads to a reduced-order solution which lacks of an accurate representation of the steep transition at the end of the time interval due to an insufficient number of time instances. In contrary, the use of the adaptive time grid leads to a better gradient representation.

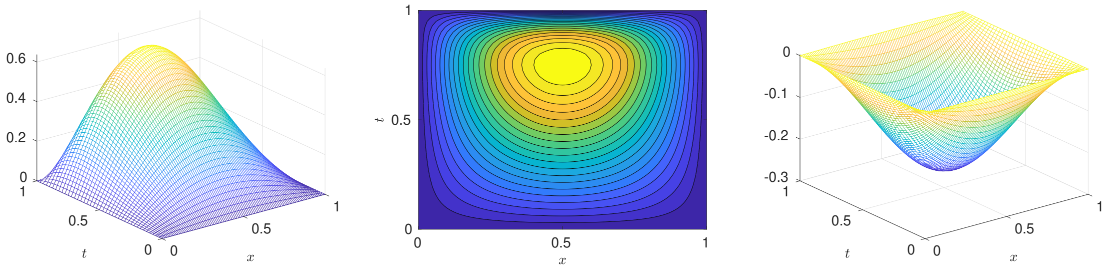

Figure 26: POD optimal adjoint state computed on an equidistant grid with $n = 40$ (left), its contour lines (middle) and the POD optimal state (right)

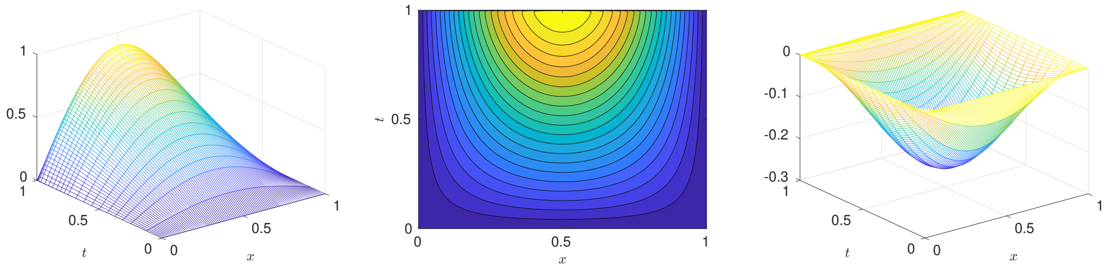

Figure 27: POD optimal adjoint state computed on an equidistant grid with $n = 40$ (left), its contour lines (middle) and the POD optimal state (right)

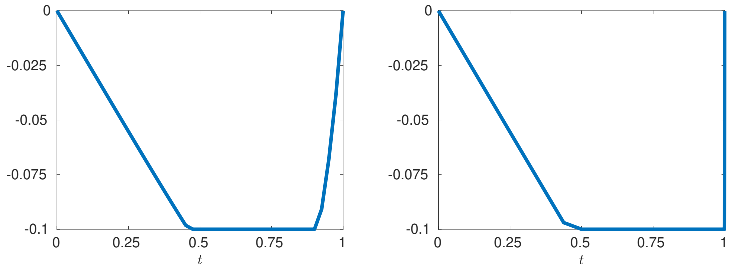

Figure 28: POD optimal controls $u_1 = u_2$ associated with a uniform (left) and an adaptive (right) time grid for the adjoint state

The relative $L^2(0,T;\Omega)$- and $L^2(0,T;\mathbb{R}^m)$-errors between the analytical optimal adjoint and control solution and the POD solution, respectively, depending on different numbers of time instances are shown in Figure 29. The inclusion of time adaptivity leads to a notable improvement of the errors in both adjoint state and the control variable.

Regarding the efficiency, we compare in Table 2 the offline and online computational times which are spent in order to achieve an accuracy of $1.20 \cdot 10^{-03}$ in the control variable. The use of an adaptive time grid for snapshot generation and POD reduced-order optimization leads to a speed up of factor 12 including all offline and online times in comparison to the use of a uniform time grid. Note that for the computation of the POD basis, we solve in the uniform and the adaptive approach an eigenvalue problem for the weighted matrix $\bar{Z}^T\bar{Z}$ which is of size $(n+1) \times (n+1)$.

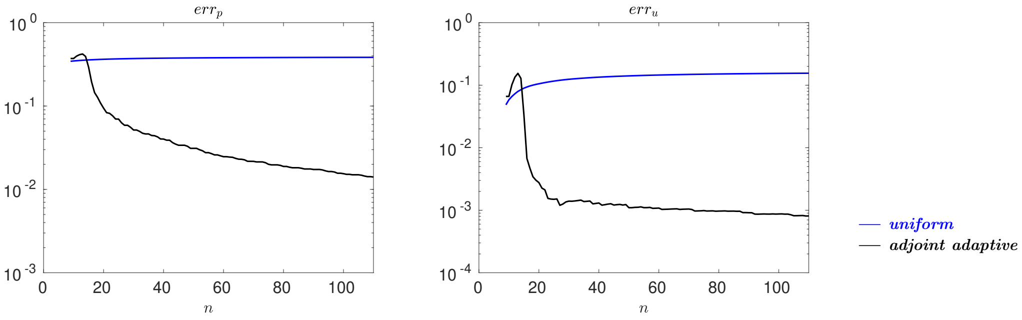

Figure 29: Relative $L^2(0,T;\Omega)$- and $L^2(0,T;\mathbb{R}^m)$-errors, respectively, between the analytical optimal solution and the POD optimal solution depending on n. Left: error in the adjoint state, right: error in the control

	uniform, FE	uniform, ROM	adaptive, ROM
n	7550	7550	40
err_u	$1.2 \cdot 10^{-03}$	$1.2 \cdot 10^{-03}$	$1.2 \cdot 10^{-03}$
optimal control solution	5.23 sec	0.58 sec	0.02 sec
snapshot generation	–	0.18 sec	0.002 sec
POD basis	–	6.34 sec	0.09 sec
time-adaptive grid	–	–	0.48 sec

Table 2: Solution of the optimal control problem using a uniform versus an adaptive time grid comparing a full finite element method with $h^* = 1/100$ with a POD reduced-order model with $\ell = 1$ POD mode

In Section 3.8.2, we have seen that the adaptive time grid according to Algorithm 9 is related to the optimal adjoint state and thus it is a suitable time grid for the adjoint state and control. However, there is no guarantee that this time grid is also suitable with respect to the state. In fact, Figure 30 shows that for this problem setting the error in the state is very similar in either of the cases of a uniform or an adaptive time mesh, although the uniform time discretization leads to slightly better results. For other problem settings, this difference might be more prominent. We follow the post-processing step according to (3.73) in order to include further time instances which are related to the state. Table 3 enumerates the improvement of the relative error in the state variable for increasing numbers of inserted time instances. The computational times for this post-processing step are very small (≤ 0.1 sec).

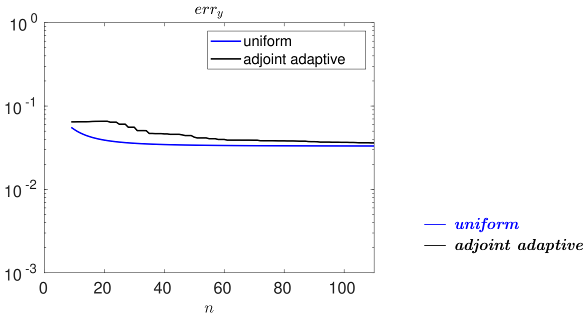

Figure 30: Relative $L^2(0,T;\Omega)$-errors between the analytical optimal state and the POD optimal state depending on n

#	err_y	time
0	$4.63 \cdot 10^{-02}$	–
5	$4.40 \cdot 10^{-02}$	0.03 sec
10	$4.31 \cdot 10^{-02}$	0.04 sec
20	$3.90 \cdot 10^{-02}$	0.07 sec
30	$3.62 \cdot 10^{-02}$	0.1 sec

Table 3: Post-processing: refinement of the adjoint adaptive grid with $n = 40$ according to (3.73) by inserting # further time instances

3.9 Further directions of research

Various modifications and extensions of the considered optimal control problem can be investigated which open new research questions. We think of the inclusion of an additional transport term, for example, or the consideration of state constraints. In this outlook, we want to give an outline to two examples of further research directions: the consideration of a nonlinear state equation and the application of the method to the model predictive control (MPC) context.

Nonlinear state equation

We consider the optimal control problem of minimizing the cost functional

$$J(y,u) = \frac{1}{2}\|y - y_d\|^2_{L^2(0,T;\Omega)} + \frac{\alpha}{2}\|u\|^2_{L^2(0,T;\Omega)}$$

subject to the semilinear parabolic equation given by

$$\begin{aligned} y_t - \Delta y + N(y) &= u \quad \text{in } (0,T] \times \Omega, && (3.74a)\\ y &= 0 \quad \text{on } [0,T] \times \partial\Omega, && (3.74b)\\ y(0,\cdot) &= y_0 \quad \text{in } \Omega. && (3.74c) \end{aligned}$$

Under certain monotonicity, boundedness and Lipschitz continuity assumptions on the nonlinearity, it is proved in e.g. [176, Theorem 5.7] that a solution to this optimal control problem exists. The associated optimality system is given by the state equation (3.74) together with the adjoint equation

$$\begin{aligned} -p_t - \Delta p + N_y(y)p &= y - y_d \quad \text{in } [0,T) \times \Omega, && (3.75a)\\ p &= 0 \quad \text{on } [0,T] \times \partial\Omega, && (3.75b)\\ p(T,\cdot) &= 0 \quad \text{in } \Omega, && (3.75c) \end{aligned}$$

and the optimality condition

$$\alpha u + p = 0 \quad \text{in } [0,T] \times \Omega. \qquad (3.76)$$

A formal reformulation of the optimality system into a biharmonic equation can now be done only with respect to the state variable in the following way. We eliminate the control in (3.74a) using (3.76) and we take the derivative with respect to time to get

$$y_{tt} - (\Delta y)_t + (N(y))_t = -\frac{1}{\alpha}p_t.$$

Utilizing (3.75a) one arrives at

$$y_{tt} - (\Delta y)_t + (N(y))_t - \frac{1}{\alpha}y = \frac{1}{\alpha}(\Delta p - N_y(y)p - y_d).$$

In order to eliminate the adjoint variable, we use (3.74a) together with (3.76). Considering the boundary conditions as well as the initial and end time condition, this leads to the nonlinear biharmonic system

$$
\begin{aligned}
-y_{tt} + \Delta^2 y + \frac{1}{\alpha} y - N_y(y)\Delta y + N_y(y)N(y) - \Delta N(y) &= \frac{1}{\alpha} y_d && \text{in } (0,T) \times \Omega, && (3.77a)\\
y &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.77b)\\
\Delta y - N(y) &= 0 && \text{on } [0,T] \times \partial\Omega, && (3.77c)\\
y(0,\cdot) &= y_0 && \text{in } \Omega, && (3.77d)\\
(y_t - \Delta y)(T) + N(y(T)) &= 0 && \text{in } \Omega. && (3.77e)
\end{aligned}
$$

It is now interesting to study the regularity of a solution to this equation depending on the nonlinearity. Moreover, since the original optimal control problem is non-convex, it is of interest to study under which conditions a solution to the optimality system is a solution to the biharmonic equation and vice versa.

Another approach is to consider a linearization of the optimal control problem. This is of interest in the context of sequential quadratic programming (SQP), for example, where a sequence of linearized problems is solved in order to find an approximate solution of the nonlinear problem. This approach is followed in [115].

Adaptive time selection in model predictive control

We aim to investigate different time selection strategies in the context of model predictive control (MPC). The principal idea in model predictive control is to split the time domain into smaller time horizons and solve corresponding open-loop control problems successively, see e.g. [88]. This approach, in particular, is applied to optimal control settings with very large or infinite (receding) time horizons. A major advantage of the approach is the possibility to react to changes of the problem data due to external influences.

A crucial challenge within the model predictive concept is the choice of the prediction horizon, i.e. the time horizon length for each finite horizon open-loop problem. Since the length of the prediction horizon strongly influences the quality and the computational times and thus the efficiency of the method, a suitable (i.e. problem-specific) choice is advantageous. We utilize the residual based time-adaptive cycle which solves the biharmonic equation iteratively and adapts the time discretization according to an error indicator. In this way, dominant temporal structures are recognized and determine the selection of appropriate time grid points and time horizon lengths. Since the resulting solution to the biharmonic system is related to the optimal solution, it can be used as a warm start in order to initialize the actual MPC iteration.

4 Adaptivity with respect to space

Challenge. In order to build a POD surrogate model for an evolution problem which reflects the spatial properties of the true solution well, the relevant spatial features need to be represented within the snapshot set. In the offline phase, the use of adaptive finite elements for snapshot generation is an efficient spatial discretization approach in order to capture important structures and guarantee an approximation of the true solution within a prescribed tolerance. In particular, many practical applications require the use of spatial adaptivity in order to make implementations feasible. As a consequence, the snapshots can have a varying number of degrees of freedom and different locations of node points. This does not fit into the usual POD framework which assumes static snapshots with the same number of degrees of freedom and location of the discretization points.

Solution idea. We provide an approach to use *adaptive* finite elements for snapshot generation in the offline phase of model order reduction utilizing proper orthogonal decomposition for semilinear parabolic evolution problems. The main idea is to consider the setting from an infinite-dimensional perspective. In particular, we assemble the snapshot gramian by evaluating inner products of snapshots explicitly. In this way, the necessity to interpolate the snapshots onto a reference space is avoided. The use of adaptivity in the offline phase for snapshot generation leads to an acceleration of the offline computation time while providing a prescribed accuracy. In the context of optimal control, the use of spatially adapted snapshots leads to a speed up in the online phase whenever new snapshots have to be generated. Moreover, for an unsteady incompressible flow problem we derive a stable POD reduced-order model by either using an optimal projection onto a weakly divergence-free space or by enriching the reduced velocity space by pressure supremizer functions.

Solution procedure. Throughout the following, we assume to have a suitable time discretization scheme at hand and concentrate on the inclusion of spatial adaptivity in the context of POD based model order reduction. We start with introducing the problem setting of a general abstract semilinear evolution equation and discretize in space using adaptive finite elements. This is done in a generic framework comprising p-, h- and r-adaptive schemes. Next, we recall the POD method in a Hilbert space setting. The continuous perspective only requires the snapshots to belong to a common Hilbert space. This is the key point of the methodological approach. The POD reduced-order model is constructed using the eigensystem of the snapshot gramian. This matrix is set up explicitly. In the case of h-adaptive finite elements with nested, hierarchical grids, the evaluation of the inner products can be done efficiently utilizing usual finite element software packages. However, in the case of r-adapted meshes, the computation of an inner product of snapshots can require the integration over cut finite elements, which is an involving task. It is realized by a detection of the cut polygon and an integration over it based on a boundary representation of the integral using Stoke's formula. The analysis for the error between the true (unknown) solution to the evolution problem and the POD solution is carried out and includes not only the POD approximation error but also the finite element error depending on the adaptive discretization scheme. In order to illustrate the feasibility of the proposed approach, we present the test case of the Cahn–Hilliard system utilizing h-adapted hierarchical meshes and two settings of a linear heat equation using nested and non-nested grids.

In the context of POD model order reduction for unsteady incompressible flows governed by the Navier–Stokes equations we propose two solution concepts in order to derive a stable reduced-order model for the case of h-adapted snapshots. The first approach leads to a velocity reduced-order model and is based on either projecting the snapshots or the POD modes onto a reference velocity space such that a weak divergence-free property is fulfilled in a common sense. The second approach yields a velocity-pressure reduced-order model. Stability is guaranteed by enriching the velocity reduced space with pressure supremizer functions. In particular, it is shown that with this construction an inf-sup condition for the reduced model holds true. We treat the inhomogeneous problem setting which requires the homogenization of the Navier–Stokes equations and a construction of a certain lifting function. Both reduced-order models are compared in a numerical test case of a lid-driven cavity flow.

Finally, we consider the problem setting of an optimal control problem governed by the Cahn–Hilliard system. The objective is to reach a desired phase distribution by controlling the velocity which enters the state equation through a transport term. We show existence of an optimal control and present a numerical test example. The novelty of the approach is the inclusion of h-adaptive finite elements for snapshot generation in an optimal control framework.

Outline of the chapter

In Section 4.1, an overview of relevant literature considering space-adaptive schemes in the context of model order reduction is given. We investigate the method of proper orthogonal decomposition for space-adapted snapshots in Section 4.2. After introducing a semilinear abstract parabolic evolution problem in Section 4.2.1, we recall the POD method in a Hilbert space framework. The construction of the snapshot gramian for space-adapted snapshots is investigated in Section 4.2.2 and a strategy for the numerical realization is presented. The method is analyzed concerning the applicability and respective advantages and disadvantages are discussed. In Section 4.3, a POD reduced-order model for the abstract semilinear evolution problem is derived for the case of space-adapted snapshots. We further discuss the treatment of nonlinearities and the expression of the POD solution in the full spatial domain. In Section 4.4 we provide an analysis for the error between the true (unknown) solution and the POD reduced-order solution utilizing space-adapted snapshots. We present in Section 4.5 a numerical example of POD reduced-order modeling for a linear heat equation and a Cahn–Hilliard model with a polynomial and a Moreau–Yosida regularized free energy using nested hierarchical adapted meshes for the snapshots. Moreover, we consider a linear heat equation using meshes with a fixed number but different locations of node points leading to non-nested grids. The numerical results for the computation of the snapshot gramian in this case using the proposed methodology of Section 4.2.2 is presented. In addition, the combination of POD model order reduction and h-adapted finite elements is considered for incompressible flow problems governed by the Navier–Stokes equations in Section 4.6. Two approaches of deriving a stable reduced-order model are proposed and compared. In particular, we propose a velocity-ROM in Section 4.6.3 which is based on an optimal projection onto a weakly divergence-free space. Further, a velocity-pressure reduced-order model is constructed in Section 4.6.4 which guarantees stability by enriching the reduced velocity space with pressure supremizer functions. The problem setting involving inhomogeneous Dirichlet and initial data is addressed in Section 4.6.5. The reduced-order models are numerically compared in the test case of a lid-driven cavity flow in Section 4.6.6. In Section 4.7, POD reduced-order modeling is used in order to find an approximate solution to an optimal control problem governed by a Cahn–Hilliard equation with transport. The existence of an optimal control is shown in Section 4.7.1. Spatially h-adapted finite element data are used for the computation of a POD basis. Finally, a numerical example is carried out in Section 4.7.2.

4.1 Literature overview

Let us give an overview of publications which consider spatial adaptivity in the context of model order reduction, presented in a chronological order.

In [72], the use of dynamically adaptive meshes is combined with POD based reduced-order modeling for an unstructured ocean model. The space-adapted snapshots are interpolated from their own mesh onto a fixed reference mesh. Afterwards, the snapshots are vectors of the same lengths and the usual POD procedure can be carried out. It is mentioned that a high-order interpolation approach can be used in order to reduce the interpolation error. The numerical

examples comprise a flow past a cylinder and a gyre.
In [126, Section 2.4.3] an interpolation approach is proposed. The main idea is to interpolate given space-adapted snapshot data (i.e. solution vectors and respective grids) by polynomials. For the evaluation of the inner products in the snapshot gramian, an appropriate numerical integration rule is used. This approach can be seen as a smoothing step, if e.g. noisy data is provided. In a numerical test in a 1d setting, snapshots with different lengths are constructed by cancelling out discretization points. A cubic spline interpolation is used and a Gauss–Legendre quadrature rule is utilized for the numerical integration. For the considered problem setting, the implementation turns out to be competitive with the standard approach on a static mesh.
In [99] POD based model reduction is considered for free boundary value problems governed by the Stefan problem. Three different numerical approaches are compared in order to generate snapshots: first a Landau-type transformation, second a control volume approach and third a moving mesh approach. For the control volume approach a rescaling and interpolation is used in order to construct discrete snapshots of the same lengths. In the moving mesh approach, the location of the grid points is varied according to the evolution of the free boundary. The grid points in the 1d setting are equidistributed and their movement is governed by an ordinary differential equation. Using a projection onto a reference domain allows to compute a POD basis in the usual manner.
In the context of model order reduction utilizing reduced basis methods for parametrized partial differential equations, an adaptive wavelet discretization is used in [7, 169] for the snapshot computations. A bound for the error between the reduced basis solution and the infinite-dimensional exact solution is derived. Moreover, a new surrogate for the exact residual and its dual norm is introduced. Convergence of the resulting adaptive greedy method is shown and numerical experiments considering a time-periodic space-time convection-diffusion-reaction example are presented.
In [45] an h-adaptive refinement approach for reduced-order models is proposed. It is an online adaptive strategy which does not require any additional full-order solutions. The main idea is to enrich the reduced space online by splitting the reduced basis vectors into separate vectors and is based on a tree structure constructed via k-means clustering. The splitting is driven by a dual weighted residual approach. In this way, a sequence of hierarchical reduced subspaces is constructed and convergence to the full-order model is shown.
The combination of POD model order reduction with h-adaptive finite element snapshots is realized in [181] by constructing common finite element spaces. Two options are considered: either all snapshots are expressed in terms of a common finite element basis or pairs of snapshots are expressed in terms of a common finite element basis of these pairs. Moreover, error estimates for a parametrized elliptic boundary value problem are proved. The approach is illustrated numerically for a parametrized convection-diffusion equation and a viscous Burgers problem.
In [190, 191, 192], a reduced basis method is developed which is certified by a dual norm bound of the residual relative to the infinite-dimensional exact solution. In the offline phase spatial mesh adaptation is used together with a greedy parameter sampling strategy leading to a reliable online system. The online bound is independent of the complexity of the underlying mixed finite element discretization and available for any parameter and not only for those which belong to the training set.

We note that Section 4.1 to Section 4.5 is based on a collaboration with Michael Hinze and is published in [84, 85]. Moreover, Section 4.6 is based on a collaboration with Michael Hinze, Jens Lang and Sebastian Ullmann. A resulting manuscript [86] is accepted for publication in Advances in Computational Mathematics in 2019. Finally, we note that Section 4.7 is based on a collaboration with Michael Hinze and Nicolas Scharmacher, is published in [87] and a further collaboration with Michael Hinze and Jan Oke Alff which is published in [6].

4.2 POD with space-adapted snapshots

This section introduces the problem setting of a general abstract semilinear parabolic evolution problem. We first apply a discretization in time which then allows us to use a different finite element space for spatial discretization at each time instance. Then, we focus on proper orthogonal decomposition in a Hilbert space setting utilizing spatially adapted snapshots. The snapshot gramian in set up explicitly by evaluating inner products of snapshots which live in a common Hilbert space. A practical implementation is proposed which realizes this approach.

4.2.1 Abstract semilinear parabolic evolution problem

Problem setting

Let us specify the abstract semilinear parabolic evolution problem which we consider in the following. Let V and H be real separable Hilbert spaces such that there exists a dense and compact embedding $V \hookrightarrow H$. The dual space H^* can be identified with H by the Riesz representation Theorem 2.8 and the Gelfand triple (V, H, V^*) is formed by

$$V \hookrightarrow H = H^* \hookrightarrow V^*, \tag{4.1}$$

see Definition 2.14. Since V is continuously embedded in H, there exists a constant $c_v > 0$ such that

$$\|v\|_H^2 \leq c_v \|v\|_V^2 \quad \text{for all } v \in V. \tag{4.2}$$

For a given symmetric, V-elliptic bilinear form $a : V \times V \to \mathbb{R}$, we assume boundedness, i.e.

$$\exists\, \beta > 0 : \quad |a(u,v)| \leq \beta \|u\|_V \|v\|_V \quad \text{for all } u, v \in V \tag{4.3}$$

and we assume that the Gårding's inequality is fulfilled (coercivity over V with respect to H), i.e.

$$\exists\, \beta_1 > 0, \beta_2 \geq 0 : \quad a(u,u) \geq \beta_1 \|u\|_V^2 - \beta_2 \|u\|_H^2 \quad \text{for all } u \in V.$$

Let $\mathcal{A} : V \to V^*$ be the bounded linear operator associated with the bilinear form a, i.e. $\mathcal{A} \in L(V, V^*)$ and

$$a(u,v) = \langle \mathcal{A}u, v \rangle_{V^*,V} = \langle u, \mathcal{A}^* v \rangle_{V,V^*} \quad \text{for all } u, v \in V,$$

where $\langle \cdot, \cdot \rangle_{V^*,V}$ denotes the dual pairing of V^* and V, see Definition 2.7, and $\mathcal{A}^*$ is the dual operator, see Definition 2.9. Moreover, we denote by $\mathcal{N} : V \to V^*$ a nonlinear operator, for which we assume local Lipschitz continuity. We are concerned with the following Cauchy problem for a semilinear evolution problem. Let $T > 0$ be a fixed end time. For a given initial function $y_0 \in H$ and external force $f \in L^2(0,T;V^*)$ we consider the problem: find $y \in W(0,T;V) \cap L^\infty(0,T;\Omega)$ with

$$\frac{d}{dt}(y(t), v)_H + a(y(t), v) + \langle \mathcal{N}(y(t)), v \rangle_{V^*,V} = \langle f(t), v \rangle_{V^*,V}, \tag{4.4}$$

and

$$(y(0), v)_H = (y_0, v)_H \tag{4.5}$$

for all $v \in V$ and for almost all $t \in (0,T]$. Note that it holds $\frac{d}{dt}(y(t), v)_H = \langle y_t(t), v \rangle_{V^*,V}$ for all $y \in L^2(0,T;V)$ with $y_t \in L^2(0,T;V^*)$ and all $v \in V$ in the sense of distributions in $(0,T)$, see e.g. [58, Chapter XVIII, §1.2, Proposition 7]. For the following, we use (4.4)-(4.5) as the governing dynamical model. We assume solvability in Assumption 4.1 and give references for a detailed analysis in Remark 4.2.

Assumption 4.1. *For given $f \in L^2(0,T;V^*)$ and $y_0 \in H$ we assume the existence of a unique solution $y \in W(0,T;V) \cap L^\infty(0,T;\Omega)$ which fulfills the initial condition* (4.5) *and solves the semilinear parabolic variational problem* (4.4)*.*

Remark 4.2. *For a detailed analysis of the existence of a unique weak solution to a semilinear parabolic evolution problem of type* (4.4)-(4.5) *we refer to [176, Lemma 5.3 and 7.12], [155, Theorem 3.1] and [48, Theorem 5.1], where an initial-boundary value problem is considered. Moreover in [189, Theorem 4.1], existence of a unique solution to a Cauchy problem for a semilinear abstract evolution equation is shown. In [145, Chapter 6] existence of solutions is discussed using semigroup methods.*

Time Discretization

In order to solve (4.4) numerically, we need to introduce a temporal and spatial discretization. As a first step, we apply a time discretization which then enables us to use a different spatial discretization for each time instance. For a given number $n \in \mathbb{N}$ of time steps let

$$0 = t_0 < \ldots < t_n = T \tag{4.6}$$

denote a time grid in the time interval $[0, T]$ with time step sizes

$$\Delta t_j = t_j - t_{j-1}$$

for $j = 1, \ldots, n$. This partition defines the time intervals $I_j = [t_{j-1}, t_j]$ for $j = 1, \ldots, n$. We follow the implicit Euler method, i.e. we integrate the differential equation (4.4) from t_{j-1} to t_j for $j = 1, \ldots, n$ and approximate the integrals involving the solution variable y by a right-sided rectangle method. Note that the integral on the right-hand side can also be approximated by the same rectangle method leading to the approximation

$$\frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \langle f(t), v \rangle_{V^*,V} dt \approx \langle f(t_j), v \rangle_{V^*,V}.$$

Of course, other integration schemes are possible. However, since f is a given function, we leave the right-hand side as it is in (4.7) at this stage in order to avoid a further error introduced by the approximation of this integral. The resulting time-discrete system reads as follows: for a given initial function $y^0 := y_0 \in H$ find solutions $y^1, \ldots, y^n \in V$ such that

$$\left(\frac{y^j - y^{j-1}}{\Delta t_j}, v \right)_H + a(y^j, v) + \langle \mathcal{N}(y^j), v \rangle_{V^*,V} = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \langle f(t), v \rangle_{V^*,V} dt \quad \text{for all } v \in V. \tag{4.7}$$

Space Discretization

At each time level, we introduce a spatial discretization for which we use adaptive finite elements. There are mainly three categories of adaptivity concepts: h-, p- and r-refinement, recall Section 2.2. All of these adaptivity schemes have the following structure in common. At each time point t_j for $j = 0, \ldots, n$, we introduce a regular triangulation $\mathcal{T}_j$ of $\bar{\Omega}$ (Definition 2.32) and define an N_j-dimensional conformal subspace V_h^j of V by

$$V_h^j := \operatorname{span}\{v_1^j, \ldots, v_{N_j}^j\} \subset V$$

with finite element basis functions $\{v_i^j\}_{i=1}^{N_j}$. For Lagrangian polynomials it holds $v_i^j(P_k^j) = \delta_{ik}$ for $i, k = 1, \ldots, N_j$ with node points $\{P_k^j\}_{k=1}^{N_j}$ of the underlying triangulation $\mathcal{T}_j$. Higher-order schemes require additionally derivative information.

At each time level $j = 0, \ldots, n$, the utilized finite element spaces $V_h^0, \ldots, V_h^n$ can differ both in the underlying triangulation of the domain $\bar{\Omega}$ and in the polynomial degree. We apply a finite element Galerkin scheme for (4.7). For this, we us denote by V_h^0 the finite element space associated with the initial triangulation $\mathcal{T}_h^0$. The fully discrete form reads as follows: find a sequence $y_h^0 \in V_h^0, \dots, y_h^n \in V_h^n$ which fulfills

$$\left(\frac{y_h^j - y_h^{j-1}}{\Delta t_j}, v \right)_H + a(y_h^j, v) + \langle \mathcal{N}(y_h^j), v \rangle_{V^*,V} = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \langle f(t), v \rangle_{V^*,V} dt, \text{ for all } v \in V_h^j \quad (4.8)$$

and

$$(y_h^0, v)_{L^2(\Omega)} = (y_0, v)_{L^2(\Omega)} \quad \forall v \in V_h^0. \quad (4.9)$$

Note that for $j = 1, \dots, n$ in (4.8) an exact evaluation of the inner product $(y_h^{j-1}, v)_H$ of the previous solution $y_h^{j-1} \in V_h^{j-1}$ and the test function $v \in V_h^j$ (i.e. an inner product of functions which live in different finite element spaces) is possible since the functions live in the common Hilbert space $V \subset H$. A possible implementation of an exact evaluation of such inner products runs analogously to the evaluation of inner products of snapshots in different finite element spaces which is explained later. Often, a Lagrange interpolation $\pi^j : C^0(\bar{\Omega}) \to V_h^j$ is used in order to interpolate the solution $y_h^{j-1} \in V_h^{j-1} \subset C^0(\bar{\Omega})$ of the previous time instance t_{j-1} to the new finite element space V_h^j. This usually simplifies the numerical implementation. However it might introduce an interpolation error which we want to avoid at this stage.
Since for $j = 0, \dots, n$ it holds $y_h^j \in V_h^j$, we make the finite element Galerkin ansatz

$$y_h^j = \sum_{i=1}^{N_j} \mathsf{y}_i^j v_i^j \in V_h^j \subset V \quad (4.10)$$

with appropriate mode coefficient vectors $\mathsf{y}^j = (\mathsf{y}_1^j, \dots, \mathsf{y}_{N_j}^j)^T \in \mathbb{R}^{N_j}$.

4.2.2 POD method with snapshots in different finite element spaces

The aim of this section is to derive a low-order POD approximation for the infinite-dimensional evolution problem (4.4) using space-adapted snapshots $y_h^0 \in V_h^0, \dots, y_h^n \in V_h^n$ with $V_h^0, \dots, V_h^n \subset V$ which are solutions to the fully discrete equations (4.8)-(4.9). The main conceptual challenge is caused by the fact that the snapshots live in different finite element spaces for each time instance. In order to understand this challenge, let us recall how the POD method works for snapshots which all belong to the same finite element space, i.e. it holds true $V_h^0 = \dots = V_h^n =: V_h$ and $y_h^0, \dots, y_h^n \in V_h \subset V$. Using the finite element Galerkin ansatz $y_h^j = \sum_{i=1}^N \mathsf{y}_i^j v_i \in V_h$ with N degrees of freedom, the equation (4.8) can be written as a matrix-vector equation for the coefficient vectors of the Galerkin ansatz as follows

$$(M + \Delta t_j A)\mathsf{y}^j = M\mathsf{y}^{j-1} + \Delta t_j b(t_j, \mathsf{y}^j), \quad (4.11)$$

where for $k, l = 1, \dots, N$ the following notations are used: $M_{kl} = (v_k, v_l)_H$, $A_{kl} = a(v_k, v_l)$ and $(b(t_j, \mathsf{y}^j))_l = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \langle f(t), v_l \rangle_{V^*,V} dt - \langle \mathcal{N}(y_h^j), v_l \rangle_{V^*,V}$. Note that each finite element coefficient vector y^j has the same dimension N. For the model order reduction of (4.11), the Galerkin finite element coefficient vectors $\{\mathsf{y}^j\}_{j=0}^n$ are taken as snapshots and a finite-dimensional POD is carried out, e.g. by solving a truncated singular value decomposition for the finite element coefficient matrix $[\mathsf{y}^0 \mid \cdots \mid \mathsf{y}^n] \in \mathbb{R}^{N \times (n+1)}$. For more details, we refer to [186, Section 1], for example.

Now, the consideration of spatially adaptive finite elements in (4.8) leads to a matrix-vector equation with time-dependent matrices and vectors as follows

$$(M^j + \Delta t_j A^j)\mathsf{y}^j = M^j \mathsf{y}^{j-1} + \Delta t_j b^j(t_j, \mathsf{y}^j), \quad (4.12)$$

where for $k, l = 1, \dots, N_j$ the following notations are used: $M^j_{kl} = (v^j_k, v^j_l)_H$, $A^j_{kl} = a(v^j_k, v^j_l)$ and $(b^j(t_j, \mathsf{y}^j))_l = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \langle f(t), v^j_l \rangle_{V^*,V} dt - \langle \mathcal{N}(y^j_h), v^j_l \rangle_{V^*,V}$. For a clear presentation, we assume at this point that the coefficient vector y^{j-1} on the right-hand side of (4.11) is the finite coefficient vector of the Lagrange interpolation of y^{j-1}_h onto the finite element space V^j_h. Since the finite element nodal basis is different for each time instance, the corresponding mode coefficient vectors $\{\mathsf{y}^j\}_{j=0}^n$ have different lengths or different meanings. In particular, in the case in which the number of node points is the same, the location of the node points might be different, for example. As a consequence, the consideration of a proper orthogonal decomposition on the level of the finite element coefficient vectors is no longer possible. The snapshot matrix cannot be set up directly and the usual POD procedure, which utilizes e.g. a singular value decomposition of the snapshot matrix, cannot be carried out.

One intuitive option to handle this issue is to interpolate all snapshots onto a reference finite element space. It is an advantage of this approach that the usual POD procedure can be followed after the interpolation. However, a major disadvantage is that interpolation errors can occur when interpolating the snapshots onto a reference finite element space. In particular, important information about the spatial structure of the snapshots might get lost, for example, if the mesh of the reference space is coarser or if the polynomial degree of the finite element ansatz functions is lower. Let us discuss advantages and disadvantages of existing approaches related to this issue.

- In order to avoid interpolation errors and ensure an exact representation of the snapshots in the reference finite element space, a common finite element space for all snapshots is constructed in [181] for the case of h-adapted snapshots which are generated using hierarchical nested grids. The construction of a common finite element mesh in this case is realized by a refinement of the initial mesh according to the adaptive cycle. In this way, the overlay mesh of all computational meshes is generated, which is the smallest possible common mesh. This overlay grid is then again a conformal triangulation of the domain and the corresponding finite element space can be set up. The snapshots are expressed with respect to the common finite element basis. This approach represents an efficient handling with usual finite element implementation tools in the case of hierarchical nested mesh structures. A possible disadvantage appears in the case in which the spatial structures differ strongly with time. Then, the common finite element space has a very large dimension. As a consequence, storage problems might occur and/or solving a singular value decomposition for the corresponding snapshot matrix might be computationally intensive. For this reason, in [181] it is also considered to construct common finite element spaces for pairs of snapshots. The presentation of this approach is restricted to h-adaptive finite elements with hierarchical nested grids.

- The construction of a common finite element space in the case of finite element snapshots with non-nested meshes is quite involved. Let us think of scenarios in which we cannot restrict the triangulation of the finite element model to nested hierarchical meshes: For example, if we receive snapshot information from physical applications or measurements or if *black-box* tools for mesh adaptation have to be used in order to work with complex domain geometries. Another situation of non-nested spatial discretizations appears in the case of r-adaptivity, where the location of node points is moved according to the spatial structures. In such cases, the overlay of the snapshot meshes might lead to cut simplices. A subtriangulation can be an option. However, this might also lead to a very fine common mesh, such that the interpolated snapshots have a very large dimension and the computation of a POD basis becomes very demanding. Moreover, as pointed out in [134] in the context of a three-dimensional example, a subtetrahedralization of an arbitrary polyhedron is challenging and additional vertices might have to be added.

- In the case of p-adapted snapshots, the underlying triangulation is fixed such that the mesh handling is not an issue. However, the number of node points and the corresponding restrictions might change according to the polynomial degree of the finite element basis functions. An interpolation of all snapshots onto the finite element space with highest polynomial degree is an option and leads to an exact representation of the snapshot ensemble without loss of information. However, it possibly contains an overhead of computation in the situation in which only a very small number of snapshots has a higher polynomial degree than the majority of the snapshots.

The goal of this section is to propose a POD reduced-order model that uses a snapshot gramian in order to construct the POD surrogate model. This approach allows to avoid the necessity of e.g. interpolating the snapshots into a reference or a common finite element space.
The crucial point of the approach is to consider the POD method from an infinite-dimensional perspective, where a finite number of snapshots are used which live in a common Hilbert space X. We can proceed along the lines of Section 2.4, *POD in Hilbert spaces.* As specified there, the only assumption that is made on the snapshots is that they need to live in a common Hilbert space, i.e. $y^0,\dots,y^n \in X$, compare (2.6). In the situation of spatially adapted snapshots this holds true, since $y_h^0 \in V_h^0,\dots,y_h^n \in V_h^n$ with $V_h^0,\dots,V_h^n \subset V$, i.e. $y_h^0,\dots,y_h^n \in V$.

As explained in Section 2.4, the idea of the POD method is to describe the space

$$\mathcal{V} := \operatorname{span}\{y_h^0,\dots,y_h^n\} \subset V \subseteq X$$

by means of few orthonormal functions $\{\psi_i\}_{i=1}^{\ell} \subset X$, with $\ell \leq d := \dim \mathcal{V} \leq \infty$, such that the error between the snapshots $\{y_h^j\}_{j=0}^n$ and the projection of the snapshots onto the subspace $X_\ell := \operatorname{span}\{\psi_1,...,\psi_\ell\} \subset X$ is minimized in the following mean-square sense

$$\min_{\psi_1,\dots,\psi_\ell \in X} \sum_{j=0}^n \alpha_j \left\| y_h^j - \sum_{i=1}^{\ell} (y_h^j,\psi_i)_X \psi_i \right\|_X^2 \quad \text{s.t.} \quad (\psi_i,\psi_j)_X = \delta_{ij} \text{ for } 1 \leq i,j \leq \ell, \tag{4.13}$$

compare (2.7), where X can denote V or H and $\{\alpha_j\}_{j=0}^n$ describe non-negative weights. For the solution of (4.13) we refer to Theorem 2.46. Instead of considering a truncated eigenvalue value problem for the operator $\mathcal{R}$ defined in (2.10) as

$$\mathcal{R} := \mathcal{Y}\mathcal{Y}^* : X \to X, \quad \psi \mapsto \sum_{j=0}^n \alpha_j (\psi, y_h^j)_X y_h^j \quad \text{for } \psi \in X,$$

we consider the snapshot gramian $\mathcal{K} := \mathcal{Y}^*\mathcal{Y} : \mathbb{R}^{n+1} \to \mathbb{R}^{n+1}$. We refer to (2.8) and (2.9) for the definition of $\mathcal{Y}$ and its Hilbert space adjoint $\mathcal{Y}^*$, respectively. The action of $\mathcal{K}$ is given by

$$\phi \mapsto \mathcal{K}\phi := \begin{pmatrix} \displaystyle\sum_{j=0}^n \sqrt{\alpha_j} (\sqrt{\alpha_0} y_h^0, y_h^j)_X \phi_j \\ \vdots \\ \displaystyle\sum_{j=0}^n \sqrt{\alpha_j} (\sqrt{\alpha_n} y_h^n, y_h^j)_X \phi_j \end{pmatrix} \quad \text{for } \phi = (\phi_0,\dots,\phi_n) \in \mathbb{R}^{n+1}.$$

The linear operator $\mathcal{K}$ can be represented as the symmetric matrix

$$\mathcal{K} = \begin{pmatrix} \sqrt{\alpha_0}\sqrt{\alpha_0}(y_h^0,y_h^0)_X & \dots & \sqrt{\alpha_0}\sqrt{\alpha_n}(y_h^0,y_h^n)_X \\ \vdots & & \vdots \\ \sqrt{\alpha_n}\sqrt{\alpha_0}(y_h^n,y_h^0)_X & \dots & \sqrt{\alpha_n}\sqrt{\alpha_n}(y_h^n,y_h^n)_X \end{pmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}. \tag{4.14}$$

We consider the eigenvalue problem of $\mathcal{K}$, i.e. we solve

$$\mathcal{K}\phi_i = \lambda_i \phi_i.$$

It can be shown that the eigenvalues $\{\lambda_i\}_{i=1}^d$ of $\mathcal{K}$ coincide with the eigenvalues for $\mathcal{R}$ except for possibly zero, see for example [121]. The corresponding orthonormal eigenvectors $\{\phi_i\}_{i=1}^d \subset \mathbb{R}^{n+1}$ for $\mathcal{K}$ are given by

$$\phi_i = \frac{1}{\sqrt{\lambda_i}}(\mathcal{Y}^* \psi_i) = \frac{1}{\sqrt{\lambda_i}} \begin{pmatrix} (\psi_i, \sqrt{\alpha_0} y_h^0)_X \\ \vdots \\ (\psi_i, \sqrt{\alpha_n} y_h^n)_X \end{pmatrix} \in \mathbb{R}^{n+1} \quad \text{for } i = 1, \dots, d.$$

This approach is also known as *method of snapshots or strobes* according to [166, p. 567]. At this point it is important to have a closer look at the snapshot gramian $\mathcal{K}$. The dimension of $\mathcal{K}$ is $(n+1) \times (n+1)$, where n denotes the number of snapshots. Thus, the matrix dimension does not depend on any spatial dimension(s) of the snapshots, it only depends on the number of utilized snapshots. In order to assemble the matrix $\mathcal{K}$, inner products of snapshots have to be computed. This is possible since $y_h^i, y_h^j \in V \subseteq X$ for $i, j = 0, \dots, n$. Since $\mathcal{K}$ is symmetric, it suffices to compute the upper (or lower) triangular matrix. A practical numerical realization of the assembly of $\mathcal{K}$ is discussed later.

The computation of the snapshot gramian $\mathcal{K}$ and its eigenvalue decomposition is sufficient in order to set up the POD reduced-order model. More details on that will be discussed in Section 4.3. Finally, the POD modes $\{\psi_i\}_{i=1}^\ell$ can be written as a linear combination of finite element functions

$$\psi_i = \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}\phi_i = \frac{1}{\sqrt{\lambda_i}} \sum_{j=0}^{n} \sqrt{\alpha_j} (\upphi_i)_j y_h^j \in V \quad \text{for } i = 1, \dots, \ell, \tag{4.15}$$

where $(\upphi_i)_j$ denotes the j-th component of $\phi_i \in \mathbb{R}^{n+1}$.

To conclude, the snapshot gramian $\mathcal{K}$ and its eigenvalue decomposition is used in order to compute the right singular vectors which sizes are independent of the spatial dimension(s) of the space-adapted finite element snapshots. Thus, this perspective makes the approach independent of the original spatial discretization method used for snapshot generation.

It remains to clarify how the snapshot gramian $\mathcal{K}$ defined in (4.14) can be set up practically. The next paragraph proposes a possible numerical implementation.

Numerical realization of the POD method utilizing snapshots from different finite element spaces

Let us now turn our perspective to the numerical realization of computing the snapshot gramian $\mathcal{K}$ defined in (4.14) for snapshots which live in different finite element spaces. For each time level $j = 0, \dots, n$, the snapshots $\{y_h^j\}_{j=0}^n$ are taken from different finite element spaces $\{V_h^j\}_{j=0}^n$ which belong to a common Hilbert space V. In the fully discrete formulation of the POD method we are given the evaluation of the snapshots on their corresponding grids, i.e. we are given the mode coefficient vectors

$$\mathsf{y}^0 \in \mathbb{R}^{N_0}, \dots, \mathsf{y}^n \in \mathbb{R}^{N_n}$$

of possibly different lengths with $\mathsf{y}^j = (\mathsf{y}_1^j, \dots \mathsf{y}_{N_j}^j)^T \in \mathbb{R}^{N_j}$, for $j = 0, \dots, n$. This is why we are not able to set up the discrete counterpart to the operator $\mathcal{R}$, which is an $N \times N$ matrix for uniform spatial discretization with N degrees of freedom.

We turn our attention to the matrix $\mathcal{K} \in \mathbb{R}^{(n+1)\times(n+1)}$. This matrix dimension only depends on the number of snapshots and can be computed for any underlying finite element discretization: the $ij-$th component $\mathcal{K}_{ij}$, for $i, j = 0, \dots, n$, is given by

$$\mathcal{K}_{ij} = \sqrt{\alpha_i}\sqrt{\alpha_j}(y_h^i, y_h^j)_X = \sqrt{\alpha_i}\sqrt{\alpha_j}\Big(\sum_{k=1}^{N_i} \mathsf{y}_k^i v_k^i, \sum_{l=1}^{N_j} \mathsf{y}_l^j v_l^j\Big)_X = \sqrt{\alpha_i}\sqrt{\alpha_j} \sum_{k=1}^{N_i} \sum_{l=1}^{N_j} \mathsf{y}_k^i \mathsf{y}_l^j (v_k^i, v_l^j)_X.$$

Thus, for any $i, j = 0, \ldots, n$, the evaluation of the inner product of two finite element snapshots y_h^i and y_h^j is traced back to the evaluation of the inner products of finite element basis functions $(v_k^i, v_l^j)_X$. Figure 31 visualizes finite element basis functions with potentially different grids and polynomial degrees. We discover, then, that the calculation of the matrix $\mathcal{K}$ as well as the determination of its eigenvectors can be done in the case of snapshots which live in different finite element spaces. Thus, the eigenvectors $\{\phi_i\}_{i=1}^d$ of $\mathcal{K}$ are the right singular vectors of $\mathcal{Y}$ and contain the space independent time information. This fact will be used in the following to build up the reduced-order model. We note that the calculation of the matrix $\mathcal{K}$ can be done for arbitrary finite element spaces, i.e. all kinds of adaptivity (h-, p- and r-adaptivity) can be considered and we do not need the interpolation onto a reference finite element space. The complexity of this methods lies in the computation of the inner products.

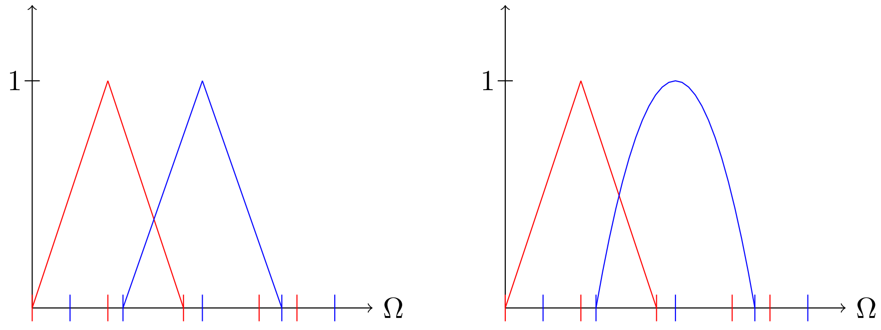

Figure 31: 1d finite element basis functions on their corresponding grids. Left: both piecewise linear finite element ansatz functions. Right: piecewise linear and cubic finite element ansatz functions

The calculation of the matrix $\mathcal{K} \in \mathbb{R}^{(n+1)\times(n+1)}$ is favorable in such a manner, since we assume the temporal dimension n to be far smaller than the spatial dimension(s) $\{N_j\}_{j=0}^n$. Due to the symmetry of $\mathcal{K}$ it suffices to compute the entries on and upon the diagonal of the matrix. The computations of the matrix entries of $\mathcal{K}$ can be done fully in parallel. Hence, if appropriate hardware is available, setting up the matrix $\mathcal{K}$ can be done fast.

Numerical realization in the case of snapshots which lead to intersecting elements

In order to explain the practical realization of an inner product of type $(y_h^i, y_h^j)_X$ of space-adapted finite element data, let us start with considering an example.

Example. We choose $V = H^1(\Omega), H = L^2(\Omega)$ and set $X = H = L^2(\Omega)$. The triangulations of $\bar{\Omega}$ for each time level $j = 0, \ldots, n$ are denoted by $\{\mathcal{T}_h^j\}_{j=0}^n$ and the finite element spaces are defined by

$$V_h^j = V(\mathcal{T}_h^j) = \{v \in C^0(\bar{\Omega}) : v|_{\mathcal{T}} \in \mathbb{P}_r(\mathcal{T}), \forall \mathcal{T} \in \mathcal{T}_h^j\} \subset X, \quad j = 0, \ldots, n,$$

where $\mathbb{P}_r$ denotes the space of polynomials of degree $r \in \mathbb{N}$. The computation of the ij-th entry $\mathcal{K}_{ij} = \sqrt{\alpha_i}\sqrt{\alpha_j}(y_h^i, y_h^j)_{L^2(\Omega)}$ of the matrix $\mathcal{K}$ is calculated by

$$\begin{aligned}\sqrt{\alpha_i}\sqrt{\alpha_j}(y_h^i, y_h^j)_{L^2(\Omega)} &= \sqrt{\alpha_i}\sqrt{\alpha_j}\int_\Omega y_h^i y_h^j dx \\ &= \sqrt{\alpha_i}\sqrt{\alpha_j}\sum_{k=1}^{N_i}\sum_{l=1}^{N_j} \mathrm{y}_k^i \mathrm{y}_l^j \int_\Omega v_k^i v_l^j dx\end{aligned}$$

$$= \quad \sqrt{\alpha_i}\sqrt{\alpha_j}\sum_{k=1}^{N_i}\sum_{l=1}^{N_j} \mathrm{y}_k^i \mathrm{y}_l^j \left(\sum_{\mathcal{T}\in\mathcal{T}_h^i} \sum_{\bar{\mathcal{T}}\in\mathcal{T}_h^j} \int_{\mathcal{T}\cap\bar{\mathcal{T}}} v_k^i v_l^j \, dx \right).$$

Computations become easier when using nested grids. In this case, the intersection of two arbitrary d-dimensional simplices coincides either with the smaller simplex, or is a common edge simplex, or has no overlap. $\Diamond$

The example reveals that the setting of dealing with different finite element meshes causes the possible challenge to deal with integrals of type

$$\int_P y_h^i y_h^j dx \tag{4.16}$$

over cut finite elements (convex polytopes) $P = \mathcal{T}\cap\bar{\mathcal{T}}$ with $\mathcal{T}\in\mathcal{T}_h^i$ and $\bar{\mathcal{T}}\in\mathcal{T}_h^j$, which involves the integration of functions defined on different (non-matching) meshes, compare Figure 32.

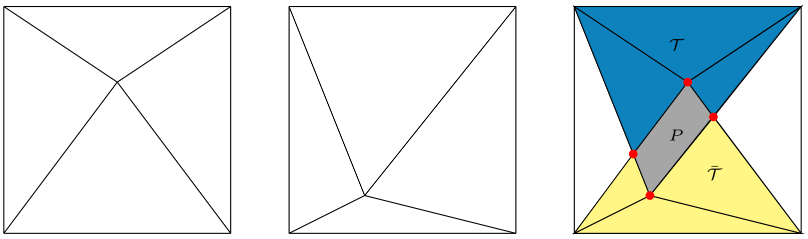

Figure 32: Two finite element meshes (left, middle) and the overlapping mesh (right)

The numerical realization is strongly based on [134], where a similar challenge is investigated in the context of multimesh methods. There, the major challenges are identified as

(i) collision detection (find intersecting simplices),

(ii) mesh intersection (detect intersection interface),

(iii) integration on complex polyhedra.

In the numerical test, we make use of built-in FEniCS [13, 132] tools to handle the issues (i) and (ii). In particular, for (i) we utilize the FEniCS collection of routines `CollisionDetection` which (amongst others) comprises an edge-edge, triangle-triangle, tetrahedron-tetrahedron collision detection and delivers the intersection coordinates for task (ii). For issue (iii) which addresses the integration over the cut elements, a subtriangulation can be an option. But as pointed out in [134] in the context of a three-dimensional example, a subtetrahedralization of an arbitrary polyhedron is challenging and additional vertices might have to be added. Moreover, the resulting subtriangulation might lead to very fine meshes such that it might slow down the performance or storage problems might occur. Therefore, an alternative approach is outlined which is based on a boundary representation of the integrals, compare [129, section 2.4] or [135] for example. Exemplarily, in the case of a two-dimensional domain with piecewise linear finite element discretization, the integrals of type (4.16) can be computed as respectively weighted sum of the integrals

$$\int_P x_1^2 dx, \quad \int_P x_2^2 dx, \quad \int_P x_1 x_2 dx, \quad \int_P x_1 dx, \quad \int_P x_2 dx, \quad \int_P 1 dx, \tag{4.17}$$

with $x = (x_1, x_2)$ using Stoke's formula

$$\int_P f(x)dx = \frac{1}{2+q}\sum_{i=1}^{m} \frac{b_i}{\|a_i\|}\int_{E_i} f(x)d\sigma_i, \tag{4.18}$$

where $\{E_i\}_{i=1}^m$ denote the edges of the polytope $P \subset \mathbb{R}^2$, $a_i^T x = b_i$ is the hyperplane in which E_i lies, and f is a polynomial of degree q. Note that the line integrals on the edges E_i can be computed using standard Gauss quadrature, for example.

It is an advantage of this approach that arbitrary snapshot data can be handled. As long as the snapshots live in the same Hilbert space, no further assumptions need to be set for the snapshots. This allows a great flexibility concerning the actual discretization scheme, such that h-, p- and r-adapted snapshots or a blend of them can be considered. Consequently, the generation of snapshot data is not a-priorily restricted to a certain discretization scheme and mesh handling. Moreover, there is no necessity of interpolating the snapshots into a reference finite element space. This means that interpolation errors can be avoided and an exact integration of the polynomials over the spatial domain is possible. The number of degrees of freedom for each snapshot is not touched and no additional degrees of freedom are introduced.

Nevertheless, the numerical implementation of integrals over cut finite elements is involving both from the computational expenses and from the implementation point of view and might need further packages additionally to the usual finite element software tools.

4.3 POD reduced-order modeling using space-adapted snapshots

In this section, we stay in the infinite-dimensional setting of the POD method and set up the POD reduced-order model utilizing snapshots which belong to different finite element spaces. This perspective allows us to determine the mode coefficients in the POD Galerkin ansatz for different underlying finite element discretizations. We start with the derivation of the POD reduced-order model for (4.4) in Section 4.3.1. The treatment of the nonlinearity is discussed in Section 4.3.2. Finally, the expression of the POD solution in full spatial dimension is outlined in Section 4.3.3.

4.3.1 POD reduced-order modeling

Suppose for given snapshots $y_h^0 \in V_h^0, \dots, y_h^n \in V_h^n$ we have computed the matrix $\mathcal{K}$ defined in (4.14), with $\mathcal{K}_{ij} = \sqrt{\alpha_i}\sqrt{\alpha_j}(y_h^i, y_h^j)_X$, for $i, j = 0, \dots, n$ as well as its eigenvalue decomposition according to the approach of the previous Section 4.2.2. Thus, we are given the ℓ largest eigenvalues $\{\lambda_i\}_{i=1}^\ell$ and corresponding eigenvectors $\{\phi_i\}_{i=1}^\ell \subset \mathbb{R}^{n+1}$ of low rank ℓ. The POD basis $\{\psi_i\}_{i=1}^\ell$ can then be written as

$$\psi_i = \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}\phi_i = \frac{1}{\sqrt{\lambda_i}} \sum_{j=0}^{n} \sqrt{\alpha_j} (\phi_i)_j y_h^j \in V \quad \text{for } i = 1, \dots, \ell,$$

according to (4.15) and thus it holds $X_\ell = \operatorname{span}\{\psi_1, \dots, \psi_\ell\} \subset V \subseteq X$. This POD basis is utilized in order to compute a reduced-order model for (4.4). For this reason, we make the POD Galerkin ansatz

$$y_\ell(t) = \sum_{i=1}^{\ell} \eta_i(t)\psi_i = \sum_{i=1}^{\ell} \eta_i(t) \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}\phi_i \quad \text{for all } t \in (0, T], \tag{4.19}$$

as an approximation for $y(t)$, with the time-dependent coefficients

$$\eta_i(t) = (y_\ell(t), \psi_i)_X = (y_\ell(t), \frac{1}{\sqrt{\lambda_i}} \mathcal{Y}\phi_i)_X$$

for $i = 1, \dots, \ell$. Inserting y_ℓ into (4.4) and choosing $X_\ell \subset V$ as the test space leads to the following problem: find $y_\ell \in X_\ell$ such that

$$\frac{d}{dt}(y_\ell(t), \psi)_H + a(y_\ell(t), \psi) + \langle \mathcal{N}(y_\ell(t)), \psi \rangle_{V^*,V} = \langle f(t), \psi \rangle_{V^*,V} \tag{4.20}$$

and

$$(y_\ell(0),\psi)_H = (y_0,\psi)_H$$

holds true for all $\psi \in X_\ell$ and for almost all $t \in (0,T]$. Utilizing the ansatz (4.19), we can write (4.20) as an ℓ-dimensional system of ordinary differential equations for the POD mode coefficients $\{\eta_i(t)\}_{i=1}^{\ell} \subset \mathbb{R}$ as follows

$$\sum_{j=1}^{\ell} \dot\eta_j(t)(\psi_j,\psi_i)_H + \sum_{j=1}^{\ell} \eta_j(t)\langle \mathcal{A}\psi_j,\psi_i\rangle_{V^*,V} + \langle \mathcal{N}(y_\ell(t)),\psi_i\rangle_{V^*,V} = \langle f(t),\psi_i\rangle_{V^*,V} \tag{4.21}$$

for $t \in (0,T], i = 1,\dots,\ell$ and

$$\sum_{j=1}^{\ell} \eta_j(0)(\psi_j,\psi_i)_H = (y_0,\psi_i)_H$$

and $i = 1,\dots,\ell$. Note that $(\psi_i,\psi_j)_H = \delta_{ij}$ if we choose $X = H$.

Since the aim is to construct a POD reduced-order model which can be built and solved for snapshots which live in different finite element spaces, we employ the results of Section 4.2.2 and rewrite system (4.21) utilizing the identity (4.15). Then, the system (4.21) can be written as

$$\begin{aligned}\sum_{j=1}^{\ell} \dot\eta_j(t)\frac{1}{\sqrt{\lambda_i}}\frac{1}{\sqrt{\lambda_j}}(\mathcal{Y}\phi_j,\mathcal{Y}\phi_i)_H + \sum_{j=1}^{\ell} \eta_j(t)\frac{1}{\sqrt{\lambda_i}}\frac{1}{\sqrt{\lambda_j}}\langle \mathcal{A}\mathcal{Y}\phi_j,\mathcal{Y}\phi_i\rangle_{V^*,V}& \\ +\frac{1}{\sqrt{\lambda_i}}\langle \mathcal{N}(y^\ell(t)),\mathcal{Y}\phi_i\rangle_{V^*,V} &= \frac{1}{\sqrt{\lambda_i}}\langle f(t),\mathcal{Y}\phi_i\rangle_{V^*,V}\end{aligned} \tag{4.22}$$

for $t \in (0,T]$ and with initial condition

$$\sum_{j=1}^{\ell} \eta_j(0)\frac{1}{\sqrt{\lambda_i}}\frac{1}{\sqrt{\lambda_j}}(\mathcal{Y}\phi_j,\mathcal{Y}\phi_i)_H = \frac{1}{\sqrt{\lambda_i}}(y_0,\mathcal{Y}\phi_i)_H.$$

In order to write (4.22) in a compact matrix-vector form, let us introduce the diagonal matrix $D \in \mathbb{R}^{\ell\times\ell}$ by

$$D = \operatorname{diag}\left(\frac{1}{\sqrt{\lambda_1}},\dots,\frac{1}{\sqrt{\lambda_\ell}}\right).$$

From the first ℓ eigenvectors $\{\phi_i\}_{i=1}^{\ell}$ of $\mathcal{K}$ we build the matrix

$$\Phi \in \mathbb{R}^{(n+1)\times\ell} \text{ by } \Phi = [\phi_1 \mid \dots \mid \phi_\ell].$$

Then, the system (4.22) can be written as the system

$$D\Phi^T\mathcal{K}\Phi D\,\dot\eta(t) + D\Phi^T\mathcal{Y}^*\mathcal{A}\mathcal{Y}\Phi D\,\eta(t) + DN(\eta(t)) = DF(t) \tag{4.23}$$

with

$$D\Phi^T\mathcal{K}\Phi D\,\eta(0) = D\bar\eta_0$$

for the vector-valued mapping $\eta(t) = (\eta_1(t),\dots,\eta_\ell(t))^T : [0,T] \to \mathbb{R}^\ell$. Note that the right-hand side $F(t)$ and the initial condition $\bar\eta_0$ can be computed as

$$(F(t))_i = \langle f(t),\mathcal{Y}\phi_i\rangle_{V^*,V} = (\mathcal{Y}^*f(t),\phi_i)_{\mathbb{R}^{n+1}}$$

and

$$(\bar\eta_0)_i = (y_0,\mathcal{Y}\phi_i)_H = (\mathcal{Y}^*y_0,\phi_i)_{\mathbb{R}^{n+1}},$$

for $i = 1, \dots, \ell$, respectively. The calculation can be done explicitly for any finite element discretization. For a given function $w \in V$ (for example $w = f(t)$ or $w = y_0$) with finite element discretization $w = \sum_{i=1}^{N_w} \mathsf{w}_i v_i$, nodal basis $\{v_i\}_{i=1}^{N_w} \subset V$ and appropriate mode coefficients $\{\mathsf{w}_i\}_{i=1}^{N_w}$ we can compute

$$(\mathcal{Y}^* w)_j = (w, y_h^j)_X = (\sum_{i=1}^{N_w} \mathsf{w}_i v_i, \sum_{k=1}^{N_j} \mathsf{y}_k^j v_k^j)_X = \sum_{i=1}^{N_w} \sum_{k=1}^{N_j} \mathsf{w}_i \mathsf{y}_k^j (v_i, v_k^j)_X, \qquad \text{for } j = 0, \dots, n$$

where $y_h^j \in V_h^j$ denotes the j-th snapshot. Again, for any $i = 1, \dots, N_w$ and $k = 1, \dots, N_j$, the computation of the inner product $(\varphi_i, v_k^j)_X$ can be done as outlined in Section 4.2.2.

For linear evolution equations the POD reduced-order model (4.23) can be set up and solved utilizing snapshots which belong to different finite element spaces, since the matrices $D\Phi^T \mathcal{K} \Phi D$ and $D\Phi^T \mathcal{Y}^* \mathcal{A} \mathcal{Y} \Phi D$ can be constructed explicitly. We discuss options in order to treat the nonlinear component $N(\eta(t))$ in the following Section 4.3.2.

Time-discrete reduced-order model

For simplicity, we choose for the reduced-order model the same temporal grid $\{t_j\}_{j=0}^n$ introduced in (4.6) which is utilized for the full-order solution. It is also possible to use a different time grid for which we refer to [121]. The time-discrete POD reduced-order model reads as follows: find a sequence $y_\ell^0, \dots, y_\ell^n \in X_\ell$ satisfying

$$\left(\frac{y_\ell^j - y_\ell^{j-1}}{\Delta t_j}, \psi \right)_H + a(y_\ell^j, \psi) + \langle \mathcal{N}(y_\ell^j), \psi \rangle_{V^*,V} = \frac{1}{\Delta t_j} \int_{t_{j-1}}^{t_j} \langle f(t), \psi \rangle_{V^*,V} dt \qquad \text{for all } \psi \in X_\ell \tag{4.24}$$

and

$$(y_\ell^0, \psi)_H = (y_0, \psi)_H$$

In matrix-vector formulation this can also be written as

$$D\Phi^T \mathcal{K} \Phi D \left(\frac{\eta^j - \eta^{j-1}}{\Delta t_j} \right) + D\Phi^T \mathcal{Y}^* \mathcal{A} \mathcal{Y} \Phi D \, \eta^j + DN(\eta^j) = DF_j \qquad \text{for } j = 1, \dots, n \tag{4.25}$$

with

$$D\Phi^T \mathcal{K} \Phi D \, \eta^0 = D\bar{\eta}_0.$$

4.3.2 Treatment of the nonlinearity

Let us now consider the computation of the nonlinear term $DN(\eta(t)) \in \mathbb{R}^\ell$ of the POD reduced-order model (4.23). It holds true

$$\begin{aligned} (DN(\eta(t)))_k &= \langle \mathcal{N}(y_\ell(t)), \psi_k \rangle_{V^*,V} \\ &= \langle \mathcal{N}(\textstyle\sum_{i=1}^\ell \eta_i(t) \psi_i), \psi_k \rangle_{V^*,V} \end{aligned}$$

for $k = 1, \dots, \ell$. It is well-known that the evaluation of nonlinearities in the context of reduced-order modeling is computationally expensive. To make this clear, let us assume we are given a uniform finite element discretization with N degrees of freedom. Then, in the fully discrete setting, the nonlinear term has the form

$$\Psi^T W \mathcal{N}(\Psi \eta(t)),$$

where $\Psi = [\psi_1 \mid \cdots \mid \psi_\ell] \in \mathbb{R}^{N \times \ell}$ denotes the matrix in which contains the POD modes columnwise and $W \in \mathbb{R}^{N \times N}$ is a weighting matrix related to the utilized inner product. Hence, the treatment of the nonlinearity requires the expansion of $y_\ell(t) = \Psi\eta(t) \in \mathbb{R}^N$ in the full space, then the nonlinearity can be evaluated and finally the result is projected back to the POD space. As a consequence, the reduced-order model is not fully independent of the high-order dimension, which slows down the simulation time - often quite drastically. Therefore, it is convenient to seek for so-called *hyper-reduction methods*, i.e. for a treatment of the nonlinearity where the model evaluation cost is related to a low dimension. Possible hyper-reduction techniques are enumerated in Section 2.4.2 in the paragraph *treatment of nonlinearities*. All the methods (except for EIM) have in common that the snapshot data needs to be available in a common reference finite element space. Since we want to circumvent this condition and set up the reduced-order model for space-adapted snapshots without the necessity of interpolation into a reference space, we have to go a different way.

We linearize and project the nonlinearity onto the POD space. For this approach, let us consider for a given initial condition $(y_\ell(0), \psi)_H = (y_0, \psi)_H$ for all $\psi \in X_\ell$ the following linear reduced-order system

$$\frac{d}{dt}(y_\ell(t), \psi)_H + a(y_\ell(t), \psi) + \langle \mathcal{N}(\bar{y}(t)), \psi \rangle_{V^*,V} = \langle f(t), \psi \rangle_{V^*,V} \quad \forall \psi \in X_\ell \tag{4.26}$$

where $\bar{y}(t) \in V$ denotes a given state. This linear evolution problem (4.26) can be set up and solved explicitly according to Section 4.3.1 without the necessity of interpolation of the snapshots into a reference finite element space. In the numerical examples, we take the finite element solution as given state in each time step, i.e. $\bar{y}(t_j) = y_h^j$.
Furthermore, the linearization of the reduced-order model (4.20) can be considered. For a given initial condition $(y_\ell(0), \psi)_H = (y_0, \psi)_H$ for all $\psi \in X_\ell$ and a given state $\bar{y}(t) \in V$ it reads as follows

$$\frac{d}{dt}(y_\ell(t), \psi)_H + a(y_\ell(t), \psi) + \langle \mathcal{N}(\bar{y}(t)) + \mathcal{N}_y(\bar{y}(t))(y_\ell - \bar{y})(t), \psi \rangle_{V^*,V} = \langle f(t), \psi \rangle_{V^*,V} \tag{4.27}$$

for all $\psi \in X_\ell$ where $\mathcal{N}_y$ denotes the Fréchet derivative. This linearized problem is of interest e.g. in the context of optimal control, where it occurs in each iteration level within the sequential quadratic programming (SQP) method, see [101, Section 2.3.2.2], for example. Choosing the finite element solution as given state in each time instance leads to:

$$\begin{aligned}
\langle \mathcal{N}(y_h^j), \psi_i \rangle_{V^*,V} &= \tfrac{1}{\sqrt{\lambda_i}} \textstyle\sum_{k=0}^n \sqrt{\alpha_k} (\phi_i)_k \langle \mathcal{N}(y_h^j), y_h^k \rangle_{V^*,V}, \\
\langle \mathcal{N}_y(y_h^j) y_\ell(t_j), \psi_i \rangle_{V^*,V} &= \langle \mathcal{N}_y(y_h^j)(\textstyle\sum_{k=1}^\ell \eta_k(t_j)\psi_k), \psi_i \rangle_{V^*,V}, \\
&= \textstyle\sum_{k=1}^\ell \eta_k(t_j) \tfrac{1}{\sqrt{\lambda_k}} \tfrac{1}{\sqrt{\lambda_i}} \sum_{p=0}^n \sum_{r=0}^n \sqrt{\alpha_p}\sqrt{\alpha_r} (\phi_k)_p (\phi_i)_r \langle \mathcal{N}_y(y_h^j) y_h^p, y_h^r \rangle_{V^*,V}, \\
\langle \mathcal{N}_y(y_h^j) y_h^j, \psi_i \rangle_{V^*,V} &= \tfrac{1}{\sqrt{\lambda_i}} \textstyle\sum_{k=0}^n \sqrt{\alpha_k} (\phi_i)_k \langle \mathcal{N}_y(y_h^j) y_h^j, y_h^k \rangle_{V^*,V}.
\end{aligned}$$

Finally, we approximate the nonlinearity $DN(\eta^j)$ in (4.25) by

$$(DN(\eta^j))_k \approx \langle \mathcal{N}(y_h^j) + \mathcal{N}_y(y_h^j)(y_\ell(t_j) - y_h^j), \psi_k \rangle_{V^*,V}$$

which can be written as

$$DN(\eta^j) \approx D\Phi^T \mathsf{N}^j + D\Phi^T \mathsf{N}_y^j \Phi D \eta^j - D\Phi^T \mathsf{N}_y^j$$

where

$$\mathsf{N}^j = \begin{pmatrix} \langle \mathcal{N}(y_h^j), \sqrt{\alpha_0} y_h^0 \rangle_{V^*,V} \\ \vdots \\ \langle \mathcal{N}(y_h^j), \sqrt{\alpha_n} y_h^n \rangle_{V^*,V} \end{pmatrix} \in \mathbb{R}^{n+1}, \quad \mathsf{N}_y^j = \begin{pmatrix} \langle \mathcal{N}_y(y_h^j) y_h^j, \sqrt{\alpha_0} y_h^0 \rangle_{V^*,V} \\ \vdots \\ \langle \mathcal{N}_y(y_h^j) y_h^j, \sqrt{\alpha_n} y_h^n \rangle_{V^*,V} \end{pmatrix} \in \mathbb{R}^{n+1}$$

and

$$\mathtt{N}_y^j = \begin{pmatrix} \langle \mathcal{N}_y(y_h^j)\sqrt{\alpha_0}y_h^0, \sqrt{\alpha_0}y_h^0\rangle_{V^*,V} & \dots & \langle \mathcal{N}_y(y_h^j)\sqrt{\alpha_n}y_h^n, \sqrt{\alpha_0}y_h^0\rangle_{V^*,V} \\ \vdots & & \vdots \\ \langle \mathcal{N}_y(y_h^j)\sqrt{\alpha_0}y_h^0, \sqrt{\alpha_n}y_h^n\rangle_{V^*,V} & \dots & \langle \mathcal{N}_y(y_h^j)\sqrt{\alpha_n}y_h^n, \sqrt{\alpha_n}y_h^n\rangle_{V^*,V} \end{pmatrix} \in \mathbb{R}^{(n+1)\times(n+1)}.$$

For weakly nonlinear systems this approximation may be sufficient, depending on the problem setting and its goal. A great advantage of linearizing the semilinear partial differential equation is that only linear equations need to be solved which leads to a further speed up. However, if a more precise approximation is desired or necessary, we can think of approximations including higher-order terms, like quadratic approximation, see e.g. [51, 157], or Taylor expansions, see e.g. [74, 148, 149]. Nevertheless, the efficiency of higher-order approximations is limited due to growing memory and computational costs.

4.3.3 Expressing the POD solution in the full spatial domain

After the computation of the solution sequence $\{\eta^j\}_{j=0}^n$ to (4.25) with $\eta^j = (\eta_1^j, \dots, \eta_\ell^j) \in \mathbb{R}^\ell$, we can write the reduced solution y_ℓ^j as

$$y_\ell^j = \sum_{i=1}^{\ell} \eta_i^j \left(\frac{1}{\sqrt{\lambda_i}} \sum_{k=0}^{n} \sqrt{\alpha_k} (\phi_i)_k y_h^k \right). \tag{4.28}$$

Let us recall the spatial discretization of the snapshots (4.10) given by

$$y_h^j = \sum_{l=1}^{N_j} \mathrm{y}_l^j v_l^j \quad \text{for } j = 0, \dots, n.$$

Now, we want to express the reduced solution y_ℓ^j in the full spatial domain. For this, we assume to be given a set of grid points $\{Q_r^j\}_{r=1}^{M_j}$ for the reduced system at time level t_j. The fully discrete POD solution can be computed by evaluation

$$y_\ell^j(Q_r^j) = \sum_{i=1}^{\ell} \eta_i^j \left(\frac{1}{\sqrt{\lambda_i}} \sum_{k=0}^{n} \sqrt{\alpha_k} (\phi_i)_k (\sum_{l=1}^{N_k} \mathrm{y}_l^k v_l^k(Q_r^j)) \right) \tag{4.29}$$

for $j = 0, \dots, n$ and $r = 1, \dots, M_j$. This allows us to use any grid for expressing the POD solution in the full spatial domain. For example, we can use the same node points at time level j for the POD simulation as we have used for the snapshots, i.e. for $j = 0, \dots, n$ it holds $M_j = N_j$ and $Q_r^j = P_k^j$ for all $r, k = 1, \dots, N_j$. Another option can be to choose

$$\{Q_r^j\}_{r=1}^{M_j} = \bigcup_{j=0}^{n} \bigcup_{k=1}^{N_j} \{P_k^j\},$$

i.e. the common grid of all snapshots. Obviously, a special and probably the easiest case concerning the implementation is to choose snapshots which are expressed with respect to a reference finite element space. Then, expression (4.29) simplifies to

$$y_\ell^j(P_r^j) = \sum_{i=1}^{\ell} \eta_i^j \left(\frac{1}{\sqrt{\lambda_i}} \sum_{k=0}^{n} \sqrt{\alpha_k} (\phi_i)_k \mathrm{y}^k \right). \tag{4.30}$$

Note that the expression of the POD solution in the full spatial domain does not belong to the POD online time (i.e. the solution of the reduced-order model). The evaluation of the reduced solution (4.29) delivers the approximated solution data needed for the application at the chosen time and space points.

4.4 Error analysis for the POD reduced-order model with space-adapted snapshots

In order to validate the accuracy of the reduced-order solution with respect to the true solution, we are interested in analyzing the quantity

$$\sum_{j=0}^{n} \alpha_j \|y(t_j) - y_\ell^j\|^2 \tag{4.31}$$

in a suitable norm, where $y(t_j)$ denotes the true (unknown) solution to (4.4) at time instances $t_0, \ldots, t_n$ and $y_\ell^j \in X_\ell$ for $j = 0, \ldots, n$ is the solution to the reduced-order model (4.24). In particular, the reduced space $X_\ell = \text{span}\{\psi_1, \ldots, \psi_\ell\}$ is constructed from snapshots $y_h^0, \ldots, y_h^n$ from different finite element spaces, i.e. $y_h^j \in V_h^j$ for $j = 0, \ldots, n$, according to Sections 4.2 and 4.3. The weights $\alpha_0, \ldots, \alpha_n$ are non-negative time weights.

Let us give references to similar error analysis studies. In [120] an a-priori estimate is proved for the error between the true solution to a nonlinear parabolic equation at certain time instances and the time-discrete reduced-order solution. In particular, [120, Theorems 7,8,9] consider an estimation for a backward Euler–Galerkin scheme, a Crank–Nicolson scheme and a Forward Euler–Galerkin scheme, respectively.
In [121, Theorem 4.7] an error estimation is proved which allows to use different time discretizations for the full-order and the reduced-order model.
Moreover, [186, Theorem 3.2.5] considers an error estimation in a finite-dimensional setting.
In order to derive an estimation for the quantity (4.31) we follow very closely the proofs in [120, 121, 186]. Here, we compute the POD basis corresponding to the fully discrete snapshots $y_h^0, \ldots, y_h^n$ using adaptive finite element spaces, which is the main differentiator to the above mentioned literature where the POD basis is computed from snapshots corresponding to the solution trajectory at given time points. For this reason, an additional term associated with the error for the spatial discretization will appear in the estimate.

We make use of the decomposition

$$y(t_j) - y_\ell^j = y(t_j) - y_h^j + y_h^j - \mathcal{P}^\ell y_h^j + \mathcal{P}^\ell y_h^j - \mathcal{P}^\ell y(t_j) + \mathcal{P}^\ell y(t_j) - y_\ell^j = \eta_j + \varrho_j + \zeta_j + \vartheta_j \tag{4.32}$$

for $j = 0, \ldots, n$, where $\eta_j := y(t_j) - y_h^j$, $\varrho_j := y_h^j - \mathcal{P}^\ell y_h^j$, $\zeta_j = \mathcal{P}^\ell y_h^j - \mathcal{P}^\ell y(t_j)$ and $\vartheta_j := \mathcal{P}^\ell y(t_j) - y_\ell^j$. The term η_j is the discretization error. We utilize the decomposition

$$\eta_j = y(t_j) - y_h^j = y(t_j) - y^j + y^j - y_h^j = E_t^j + E_h^j$$

where y^j denotes the solution to the time-discrete problem (4.7). By $E_t^j := y(t_j) - y^j$ we denote the global time discretization error and $E_h^j := y^j - y_h^j$ is the global spatial discretization error. It is

$$\|E_h^j\|_H \leq \max_{j=0,\ldots,n} \|E_h^j\|_H =: \varepsilon_h$$

and

$$\|E_t^j\|_H \leq \max_{j=0,\ldots,n} \|E_t^j\|_H =: \varepsilon_t.$$

Since we use the implicit Euler method for time integration, it is $\varepsilon_t = \mathcal{O}(\Delta t)$ with $\Delta t := \max_{j=0,\ldots,n} \Delta t_j$. Therefore, we can estimate

$$\sum_{j=0}^{n} \alpha_j \|y(t_j) - y_h^j\|_H^2 \leq \sum_{j=0}^{n} \alpha_j \|E_t^j + E_h^j\|_H^2 \leq 2 \sum_{j=0}^{n} \alpha_j ((\Delta t)^2 + \varepsilon_h^2) \leq 2T((\Delta t)^2 + \varepsilon_h^2). \tag{4.33}$$

Moreover, we have

$$\sum_{j=0}^{n} \alpha_j \|\zeta_j\|_H^2 = \sum_{j=0}^{n} \alpha_j \|\mathcal{P}^\ell y_h^j - \mathcal{P}^\ell y(t_j)\|_H^2 \leq \|\mathcal{P}^\ell\|_{\mathcal{L}(H)}^2 \sum_{j=0}^{n} \alpha_j \|\eta_j\|_H^2. \tag{4.34}$$

The term ϱ_j is the projection error of the snapshot y_h^j projected onto the POD space V_ℓ. Using (4.2), the weighted sum of all projection errors is given by the sum of the neglected eigenvalues (2.12), i.e.

$$\sum_{j=0}^{n} \alpha_j \|\varrho_j\|_H^2 = \sum_{j=0}^{n} \alpha_j \|y_h^j - \sum_{i=1}^{\ell} (y_h^j, \psi_i)_V \psi_i\|_H^2 \leq c_v \sum_{j=0}^{n} \alpha_j \|y_h^j - \sum_{i=1}^{\ell} (y_h^j, \psi_i)_V \psi_i\|_V^2 \leq c_v \sum_{i=\ell+1}^{d} \lambda_i. \tag{4.35}$$

It remains to estimate the term ϑ_j which is the error between the projection of the true solution $y(t_j)$ at time instance t_j onto the POD space V_ℓ and the time-discrete ROM solution y_ℓ^j to (4.24). With the use of the notation $\bar{\partial}\vartheta_j = (\vartheta_j - \vartheta_{j-1})/\Delta t_j$ for $j = 1, \ldots, n$, we get

$$\begin{aligned}
(\bar{\partial}\vartheta_j, \psi)_H &= (\mathcal{P}^\ell \left(\frac{y(t_j) - y(t_{j-1})}{\Delta t_j} \right) - \frac{y_\ell^j - y_\ell^{j-1}}{\Delta t_j}, \psi)_H \\
&= (\mathcal{P}^\ell \left(\frac{y(t_j) - y(t_{j-1})}{\Delta t_j} \right) + \mathcal{N}(y_\ell^j) - f_j, \psi)_H + a(y_\ell^j, \psi) \\
&= (\mathcal{P}^\ell \left(\frac{y(t_j) - y(t_{j-1})}{\Delta t_j} \right) - \frac{y(t_j) - y(t_{j-1})}{\Delta t_j} + \mathcal{N}(y_\ell^j) - \mathcal{N}(y(t_j)), \psi)_H + a(y_\ell^j - y(t_j), \psi) \\
&= (z_j + \mathcal{N}(y_j^\ell) - \mathcal{N}(y(t_j)), \psi)_H + a(y_\ell^j - y(t_j), \psi)
\end{aligned}$$

for $\psi \in V_\ell$ with $z_j := \mathcal{P} \left(\frac{y(t_j) - y(t_{j-1})}{\Delta t_j} \right) - \frac{y(t_j) - y(t_{j-1})}{\Delta t_j}$. With the choice $\psi = \vartheta_j$ and the use of the identity

$$2\langle u - v, u \rangle = \|u\|^2 - \|v\|^2 + \|u - v\|^2$$

we obtain

$$\|\vartheta_j\|_H^2 \leq \|\vartheta_{j-1}\|_H^2 + 2\Delta t_j \left(\beta \|y_\ell^j - y(t_j)\|_H + \|z_j\|_H + \|\mathcal{N}(y_\ell^j) - \mathcal{N}(y(t_j))\|_H \right) \|\vartheta_j\|_H,$$

where we have utilized (4.3). We assume that $\mathcal{N}$ is Lipschitz continuous, i.e. there exists $L > 0$ such that

$$\|\mathcal{N}(y_j^\ell) - \mathcal{N}(y(t_j))\|_H \leq L \|y_\ell^j - y(t_j)\|_H \quad \text{for } j = 0, \ldots, n.$$

Applying Young's inequality we find

$$\|\vartheta_j\|_H^2 \leq \|\vartheta_{j-1}\|_H^2 + \Delta t_j (c_1 \|\varrho_j\|_H^2 + c_2 \|\vartheta_j\|_H^2 + \|z_j\|_H^2 + c_1 \|\eta_j\|_H^2 + c_1 \|\zeta_j\|_H^2)$$

with the constants $c_1 := \beta + L$ and $c_2 := 5(\beta + L) + 1$. Under the assumption that Δt is sufficiently small, we conclude

$$\|\vartheta_j\|_H^2 \leq e^{2c_2 j \Delta t} \left(\|\vartheta_1\|_H^2 + \sum_{k=1}^{j} \Delta t_k (\|z_k\|_H^2 + c_1 \|\varrho_k\|_H^2 + c_1 \|\eta_k\|_H^2 + c_1 \|\zeta_k\|_H^2) \right). \tag{4.36}$$

For more details on this, we refer to [121, 186]. We choose the initial condition for (4.24) such that $\vartheta_0 = \mathcal{P}^\ell y(t_0) - y_0^\ell = \mathcal{P}^\ell g - y_0^\ell = 0$.

Next, we estimate the term involving z_k. It holds true

$$\begin{aligned}\|z_k\|_H^2 &= \|\mathcal{P}^\ell\left(\frac{y(t_k)-y(t_{k-1})}{\Delta t_k}\right)-\frac{y(t_k)-y(t_{k-1})}{\Delta t_k}\|_H^2\\ &= \|\mathcal{P}^\ell\left(\frac{y(t_k)-y(t_{k-1})}{\Delta t_k}\right)-\mathcal{P}^\ell\dot y(t_k)+\mathcal{P}^\ell\dot y(t_k)-\dot y(t_k)+\dot y(t_k)-\frac{y(t_k)-y(t_{k-1})}{\Delta t_k}\|_H^2\\ &\leq 2\|\mathcal{P}^\ell\|_{\mathcal{L}(H)}^2\|\frac{y(t_k)-y(t_{k-1})}{\Delta t_k}-\dot y(t_k)\|_H^2+2\|\mathcal{P}^\ell\dot y(t_k)-\dot y(t_k)\|_H^2+2\|\dot y(t_k)-\frac{y(t_k)-y(t_{k-1})}{\Delta t_k}\|_H^2\\ &\leq c_3\|w_k\|_H^2+2\|\mathcal{P}^\ell\dot y(t_k)-\dot y(t_k)\|_H^2\end{aligned}$$

with $c_3 = 2+2\|\mathcal{P}^\ell\|_{\mathcal{L}(H)}^2$ and $w_k := \dot y(t_k)-\dfrac{y(t_k)-y(t_{k-1})}{\Delta t_k}$, which can be estimated as

$$\sum_{k=1}^j \Delta t_k\|w_k\|_H^2 \leq \frac{(\Delta t)^2}{3}\|\ddot y\|_{L^2(0,t_j;H)}^2.$$

For more details on this, we refer to [121, 186].
Finally, we can summarize the estimation for the term involving ϑ_j by

$$\|\vartheta_j\|_H^2 \leq c_4\left(\sum_{k=0}^n \alpha_k(\|\mathcal{P}^\ell\dot y(t_k)-\dot y(t_k)\|_H^2+\|\varrho_k\|_H^2+\|\eta_k\|_H^2+\|\zeta_k\|_H^2)+(\Delta t)^2\|\ddot y\|_{L^2(0,T;H)}^2\right)$$

with $c_4 := e^{2c_2T}\max\{\frac{c_3}{3},4,2c_1\}$ and thus it is

$$\begin{aligned}\sum_{j=0}^n\alpha_j\|\vartheta_j\|_H^2 \quad\leq\quad & c_4T(\sum_{j=0}^n\alpha_j\|\mathcal{P}^\ell\dot y(t_j)-\dot y(t_j)\|_H^2+(\Delta t)^2\|\ddot y\|_{L^2(0,T;H)}^2+\sum_{i=\ell+1}^d\lambda_i\\ &+2T(1+\|\mathcal{P}^\ell\|_{\mathcal{L}(H)}^2)((\Delta t)^2+\varepsilon_h^2)).\end{aligned} \tag{4.37}$$

Theorem 4.3. *Let $y(t_0),\dots,y(t_n)$ denote the solution to problem (4.4) at the time instances $t_0,\dots,t_n$ and let $y_\ell^0,\dots,y_\ell^n$ be the solution to (4.24). Let the nonlinear operator $\mathcal{N}$ be Lipschitz continuous with Lipschitz constant L and let the maximal time step $\Delta t := \max_{j=0,\dots,n}\Delta t_j$ be sufficiently small. Furthermore, we assume $\ddot y(t)$ to be bounded on $[0,T]$. We choose the initial condition for (4.24) such that $\mathcal{P}^\ell g = y_\ell^0$ is fulfilled. Then, there exists a constant $C = C(T,c_v,\|P^\ell\|_{\mathcal{L}(H)}^2,\beta,L,\|\ddot y\|_{L^2(0,T,H)}^2) > 0$ such that*

$$\sum_{j=0}^n\alpha_j\|y(t_j)-y_\ell^j\|_H^2 \leq C\left((\Delta t)^2+\varepsilon_h^2+\sum_{i=\ell+1}^d\lambda_i+\sum_{j=0}^n\alpha_j\|\mathcal{P}^\ell\dot y(t_j)-\dot y(t_j)\|_H^2\right), \tag{4.38}$$

where c_v,β are from (4.2),(4.3), the quantity $\varepsilon_h := \max_{j=0,\dots,n}\|y^j-y_h^j\|_H$ refers to the global spatial discretization error and y^j is the solution to (4.7) at time instance t_j for $j=0,\dots,n$.

Proof. Utilizing the decomposition (4.32) we infer

$$\sum_{j=0}^n\alpha_j\|y(t_j)-y_\ell^j\|_H^2 \leq 2\sum_{j=0}^n\alpha_j(\|\eta_j\|_H^2+\|\varrho_j\|_H^2+\|\zeta_j\|_H^2+\|\vartheta_j\|_H^2).$$

Together with (4.33), (4.34), (4.35) and (4.37) this leads to the claim. □

Remark 4.4. *i) Note that* (4.38) *in Theorem 4.3 depends on an error of time derivatives. As discussed in [103, Remark 4.8(b)] this dependency can be avoided by enriching the snapshot set with time derivatives (more specifically, finite difference approximations of time derivatives), see also [102, 121].*
Note that for a specific setting it is possible to avoid to include time derivatives into the snapshot set. This is based on [165]. A comprehensive analysis on whether snapshot difference quotients are needed in POD is given in [109].

ii) If we choose $V = H^1(\Omega)$ and $H = L^2(\Omega)$ and utilize a static piecewise linear finite element discretization with h being the diameter of the triangles, then $\varepsilon_h = \mathcal{O}(h^2)$. It is possible to control the spatial discretization error a-posteriori using residual based error estimations.

4.5 Numerical examples

We consider three numerical examples in order to quantify numerically how the inclusion of spatial adaptivity in the offline phase for snapshot generation in POD model order reduction effects the efficiency and accuracy of the reduced-order model in comparison to the use of a static and uniform spatial discretization.

In the first two examples we focus on structured, hierarchical and nested meshes and utilize an h-adaptive finite element concept which controls the spatial discretization error in the energy norm. The first example (Section 4.5.1) is a linear heat equation and the second example (Section 4.5.2) is concerned with a phase field system governed by a Cahn–Hilliard system with transport. In both cases, we use conformal, piecewise linear and continuous finite elements and follow the usual adaptive cycle given by

solve - estimate - mark - refine/coarsen,

recall Algorithm 1. As an error indicator we utilize at each time instance the jump of the normal derivative across the edges given by

$$\sum_{E\in\mathcal{E}} h_E \|\mathbb{J}_E(\vec{n}_E \cdot \nabla y_h^j)\|^2_{L^2(E)}, \tag{4.39}$$

where h_E is the length of the edge, $\mathbb{J}_E$ denotes the jump of the function across the edge E (see Definition 2.35), $\vec{n}_E$ is the outward normal derivative on the edge E, y_h^j is the finite element solution at the current time and $\mathcal{E}$ denotes the collection of all (interior) edges in the current triangulation. It is shown in [47] that the edge residuals dominate a-posteriori error estimates for low-order finite element methods. For this reason, we neglect the further terms within residual based error indicators. For reasons of efficiency, we note that at each time instance (except for the first one) we perform only one refinement and coarsening step, since we expect only minor dynamical changes from one time instance to another assuming that the time step size is small enough.

These numerical test cases illustrate our approach explained in Sections 4.2 and 4.3 to set up and solve a POD reduced-order model utilizing space-adapted snapshots in the specific case in which a nested mesh structure is at hand. Thus, the computations benefit from the fact that the intersection of two triangles coincides either with the smaller triangle or is a common edge or has no overlap. We exploit this property in the numerical implementation. The results are compared to the use of a uniform mesh, where the mesh size coincides with the fineness of the smallest triangle in the adaptive meshes. The aim of this comparison is to investigate numerically how the inclusion of spatial adaptivity for the snapshot discretization affects the accuracy and efficiency of the POD reduced-order solution compared to using a static mesh without spatial adaptivity. Furthermore, we compare the approach to the use of a common reference mesh following [181]. For this, the practical numerical concept works as follows. In a full-dimensional simulation, h-adaptive snapshots are generated. At the same time we carry along a reference grid with the simulation, which coincides with the computational grid at initial time and is only refined (not coarsened) in the same manner as the computational grid. In this way, the reference grid becomes the overlay of all computational grids at the end of the snapshot generation. Then, the snapshots are expressed with respect to the finite element basis corresponding to the overlay mesh and the usual POD procedure is carried out.

In the POD framework, we choose $X = L^2(\Omega)$ as Hilbert space in the setting of (2.7).

The third example (Section 4.5.3) focuses on the realization of the numerical computation of the correlation matrix $\mathcal{K}$ defined in (4.14), as described in Section 4.2.2 for non-nested meshes. In this case, the overlap of two meshes leads to cut finite elements, which are convex polytopes of more than three node points such that a boundary representation of the integrals is utilized.

All coding is done in `C++`, we utilize FEniCS [13, 132] for the solution of the differential equations utilizing finite elements and ALBERTA [161] for dealing with hierarchical meshes. We run the numerical tests on a compute server with 1536 GB RAM. We note that we use libraries from [114] for the coupling of FEniCS with ALBERTA for mesh refinement and coarsening.

4.5.1 Linear heat equation

As an example for the evolution problem (4.4) we consider a linear heat equation. In the setting of the Gelfand triple (4.1) we choose $V = H_0^1(\Omega)$ and $H = L^2(\Omega)$ as Hilbert spaces. For a given source function $f \in L^2(0,T;V^*)$ and initial condition $g \in L^2(\Omega)$, we consider the following linear heat equation with homogeneous Dirichlet boundary conditions

$$\begin{aligned} y_t - \Delta y &= f \quad \text{in } (0,T] \times \Omega, && (4.40a)\\ y &= 0 \quad \text{on } [0,T] \times \partial\Omega, && (4.40b)\\ y(0,\cdot) &= g \quad \text{in } \Omega. && (4.40c) \end{aligned}$$

We can write (4.40) as an abstract evolution problem of type (4.4) by deriving a variational formulation for (4.40) with the bilinear form $a : V \times V \to \mathbb{R}$ introduced by

$$a(u,v) = (\nabla u, \nabla v)_{L^2(\Omega)}.$$

Problem setting

The spatial domain is chosen as the open unit square $\Omega = (0,1) \times (0,1) \subset \mathbb{R}^2$ and the end time is set to $T = 2$. We construct an example in such a way that we know the analytical solution. It is given by

$$\bar{y}(t,x) = r(t,x) \cdot (s_1(t,x) - s_2(t,x))$$

with

$$r(t,x) = \frac{50000 \cdot x_1 \cdot (1-x_1) \cdot (0.5+\cos(t)\cdot(x_1-0.5)-\sin(t)\cdot(x_2-0.5))^4 \cdot \frac{1}{t+1} \cdot (1-(0.5+\cos(t)\cdot(x_1-0.5)-\sin(t)\cdot(x_2-0.5)))^4)}{1+1000\cdot(\cos(t)\cdot(x_1-0.5)-\sin(t)\cdot(x_2-0.5))^2},$$

$$s_1(t,x) = \frac{10000 \cdot x_2 \cdot (1-x_2) \cdot (0.5+\sin(t)\cdot(x_1-0.5)+\cos(t)\cdot(x_2-0.5))^2 \cdot (0.5-\sin(t)\cdot(x_1-0.5)-\cos(t)\cdot(x_2-0.5))^2}{1+100\cdot((0.5+\sin(t)\cdot(x_1-0.5)+\cos(t)\cdot(x_2-0.5))-0.25)^2},$$

$$s_2(t,x) = \frac{10000 \cdot x_2 \cdot (1-x_2) \cdot (0.5+\sin(t)\cdot(x_1-0.5)+\cos(t)\cdot(x_2-0.5))^2 \cdot (0.5-\sin(t)\cdot(x_1-0.5)-\cos(t)\cdot(x_2-0.5))^2}{(1+100\cdot((0.5+\sin(t)\cdot(x_1-0.5)+\cos(t)\cdot(x_2-0.5))-0.75)^2},$$

where we use the notation $x = (x_1, x_2)$. The forcing term f and the initial condition g are chosen accordingly. For the temporal discretization we introduce the uniform time grid by

$$t_j = j\Delta t$$

for $j = 0, \ldots, 2000$ with time step size $\Delta t = 0.001$.

Finite element results

We start the adaptive finite element Algorithm 1 with an initial triangulation $\mathcal{T}_h^{init}$ of the spatial domain Ω which we choose as a structured criss-cross mesh with 20 node points in both x_1- and x_2-direction. For the Dörfler marking in Definition 2.34, we set $\theta = 0.9$ and utilize a coarsening

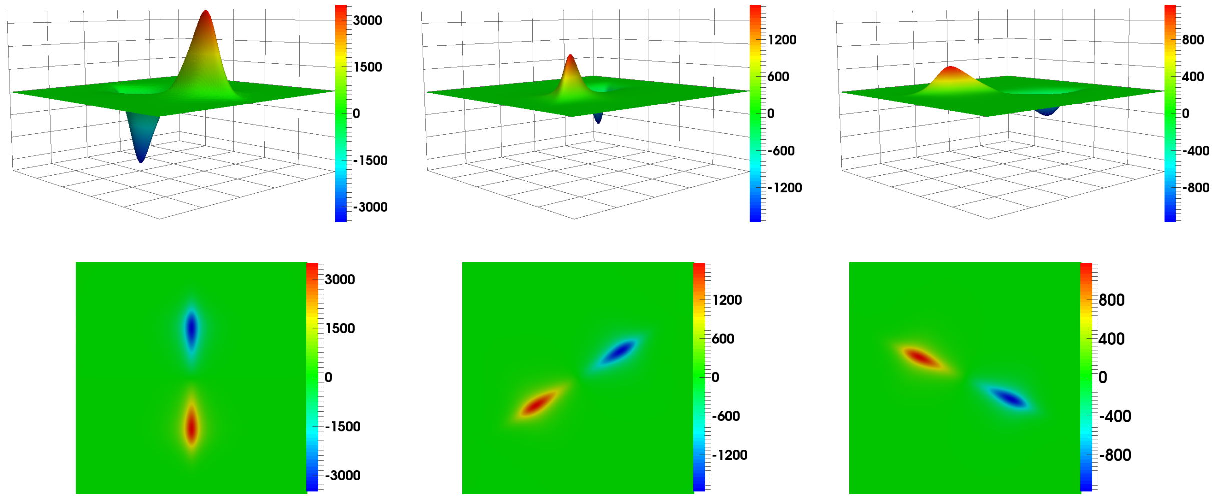

Figure 33: Surface plot (top) and top view (bottom) of the adaptive finite element solution of (4.40) at $t = t_0$ (left), $t = T/2$ (middle) and $t = T$ (right)

strategy according to [114, Section 8.3] with tolerance $\theta^c = 0.06$. A surface plot of the finite element solution at three different time points is shown in Figure 33.
Due to the steep gradients in the neighborhood of the minimum and maximum, respectively, the use of an adaptive finite element discretization is justified. A top view of the adaptive finite element solutions as well as the adapted finite element meshes and the overlay mesh are shown in Figure 34. The number of node points of the adaptive meshes varies between 4284 and 13076 points. The finest mesh has 42997 node points. In contrary, a uniform mesh with the same discretization fineness as the smallest triangle in the adaptive grids with $h_{\min} = 0.0032$ has 185437 node points. This clearly reveals the benefit of using adaptive meshes for snapshot generation. In particular, the comparison of the computational times emphasizes the benefit of adaptive snapshot sampling: the snapshot generation utilizing adaptive meshes takes on average 1053 seconds (17.5 minutes), whereas utilizing the uniform mesh takes around 14143 seconds (3.9 hours). Therefore, the inclusion of spatial adaptivity leads to a speed up factor of 13.4 in the offline phase (see Table 4) for snapshot generation.

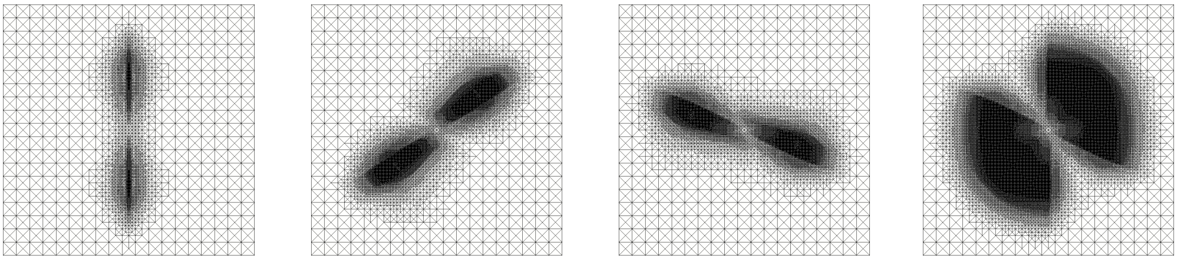

Figure 34: From left to right: Adapted finite element meshes at $t = t_0$, $t = T/2$, $t = T$ and the overlay mesh

POD reduced-order results

We investigate the inclusion of spatial adaptivity for snapshot generation in comparison to the use of a static uniform mesh. In Figure 35, the decay of the normalized eigenspectrum of the snapshot gramian for uniform spatial discretization is shown in contrast to the normalized eigenvalues using h-adaptive finite elements with nested hierarchical meshes. We note that about the first 40 eigenvalues computed corresponding to space-adapted snapshots coincide with eigenvalues associated with snapshots from a simulation on a uniform mesh. Then, the methods deliver different results: for the uniform discretization the normalized eigenvalues fall below machine precision and stagnate. In contrary, the normalized eigenvalues for the adaptive approach flatten in the order around 10^{-10}. If the marking tolerance θ for the Dörfler marking is chosen larger, i.e. $\theta = 0.98$ (or smaller, i.e. $\theta = 0.5$), the stagnation of the eigenvalues in the adaptive method takes place at a higher (or lower) order (Figure 35, right).

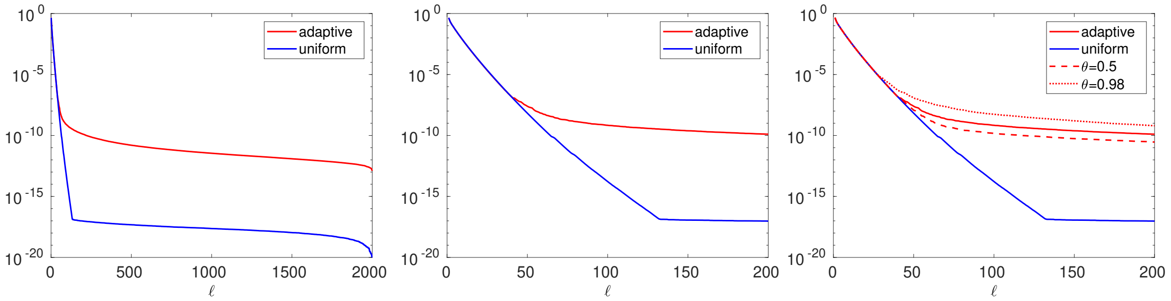

Figure 35: Decay of the normalized eigenvalues utilizing an h-adaptive and a static finite element discretization, respectively. Left: all eigenvalues, middle: first 200 largest eigenvalues, right: first 200 largest eigenvalues with different Dörfler marking tolerances θ for the adaptivity

Concerning dynamical systems, the magnitude of the eigenvalue corresponds to the dominant characteristic properties of the underlying dynamical system: the larger the eigenvalue, the more information is contained in the corresponding eigenfunction. Since all adaptive meshes are contained in the uniform mesh, the difference in the amplitude of the eigenvalues is due to the interpolation errors during coarsening. This is the price we have to pay in order to get a fast snapshot generation utilizing adaptive finite elements. Moreover, the investigation of the decay of the eigenvalues can be interpreted as an analyzing tool for adaptivity in the following sense: using an adaptive mesh technique means that some parts of the domain are resolved coarsely according to the utilized error estimation, i.e. information gets lost compared to a uniform fine resolution. In the sense of a singular value analysis, this can be explained that adaptivity neglects the noise which is indicated by the singular values on the uniform spatial mesh at those places which are not resolved with the adaptive grid. We conclude that the overtones which get lost in the adaptive computations lie in the same space which is not considered by POD when using the adaptive finite element snapshots. This allows us to characterize the space which is not resolved by adaptivity. From this point of view, adaptivity can be interpreted as a smoother.

Since the first few POD modes are the most important ones regarding the captured information (*coherent structures*), we visualize ψ_1, ψ_2 and ψ_5 in Figure 36, which are computed from space-adapted snapshots. Note that the POD bases are unique up to the sign. We can recognize the initial condition in the first POD basis function. Then, the index of the POD basis corresponds to the number of maxima and minima of the POD basis: ψ_2 has two minima and two maxima etc. This behavior is similar to the increasing oscillations in higher frequencies within trigonometric approximations. The increasing number of oscillations is necessary in order to approximate the transport of the steep gradients of the solution with increasing accuracy.

The solutions to the POD reduced-order model with dimensions $\ell = 10$ and $\ell = 40$, respectively, are shown in Figure 37, where space-adapted snapshots are utilized. The visual comparison reveals that less oscillations appear in the POD solution, if the number of utilized POD basis functions is increased (until stagnation of the corresponding eigenvalues).

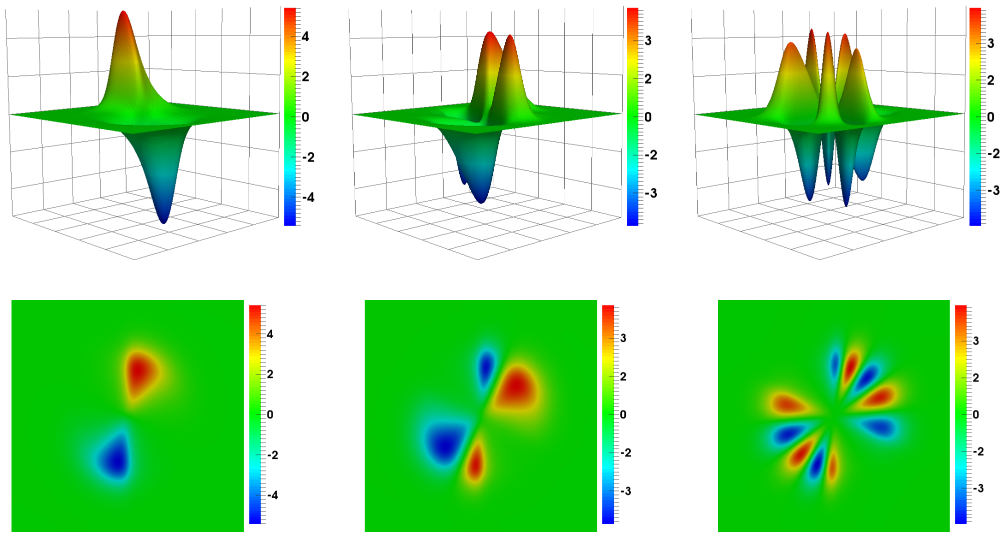

Figure 36: Surface plot (top) and top view (bottom) of the POD basis functions ψ_1 (left), ψ_2 (middle) and ψ_5 (right)

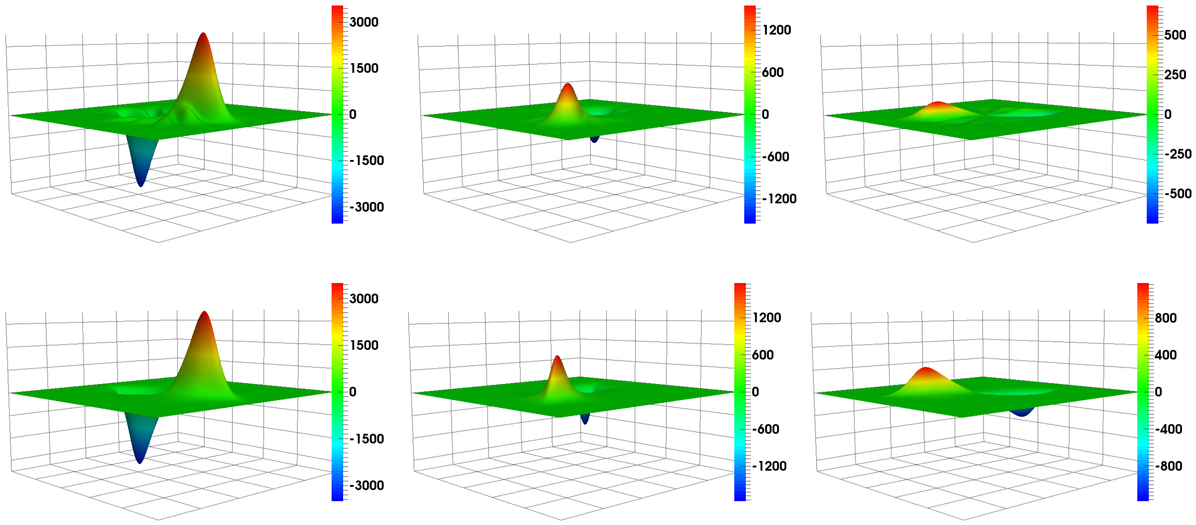

Figure 37: Surface plot of the POD reduced-order solution utilizing $\ell = 10$ (top) and $\ell = 40$ (bottom) POD modes at $t = t_0$ (left), $t = T/2$ (middle) and $t = T$ (right)

In order to validate the approximation quality of the POD reduced-order solution we consider the following relative errors

i) relative $L^2(0,T;\Omega)$-error between the finite element solution and the POD reduced-order solution given by

$$err_{FE-POD} = \frac{\|y_h - y_\ell\|_{L^2(0,T;\Omega)}}{\|y_h\|_{L^2(0,T;\Omega)}}$$

ii) relative $L^2(0,T;\Omega)$-error between the true solution and the POD reduced-order solution given by

$$err_{true-POD} = \frac{\|\bar{y} - y_\ell\|_{L^2(0,T;\Omega)}}{\|\bar{y}\|_{L^2(0,T;\Omega)}}$$

iii) POD projection error given by

$$err_{PODproj} = \frac{\|y_h - \mathcal{P}_H^\ell y_h\|_{L^2(0,T;\Omega)}}{\|y_h\|_{L^2(0,T;\Omega)}} = \frac{\sqrt{\sum_{i>\ell} \lambda_i}}{\sqrt{\sum_i \lambda_i}}$$

Figure 38 illustrates these errors (i)-(iii) utilizing static versus space-adapted snapshots depending on different POD basis lengths. Let us start with analyzing the decay of the relative error in the case of uniform and static meshes (Figure 38, right). In this case, the POD reduced-order solution converges to the finite element solution for an increasing number of utilized POD modes. In fact, if we utilize $\ell = \dim(\text{span}\{y_h^1, \dots, y_h^n\})$ POD modes, then the reduced space coincides with the finite element space. The error between the finite element solution and the POD reduced-order solution is very close to the POD projection error which is the lower bound for the error err_{FE-POD}. However, the error between the POD reduced-order solution and the true solution $\bar{y}$ stagnates in the order of 10^{-03} for $\ell \geq 40$, since the finite element discretization error then dominates the overall error according to Theorem 4.3. This is in accordance to the decay of the eigenvalues shown in Figure 35.

Let us now look at the relative errors with regard to the space-adapted snapshots (Figure 38, middle). We also observe a stagnation of the error between the true solution and the POD reduced-order solution. In particular, the stagnation starts earlier and is at an order of 10^{-02}, i.e. larger than in the static case. This is due to the fact that the finite element discretization error is larger in the case of space-adapted solutions, since we are concerned with adapted meshes. Moreover, the POD projection error stagnates in the case of space-adapted snapshots. This is in accordance to the decay of the eigenvalues (compare Figure 35). It is related to the finite element error in the snapshots. For an increasing number of utilized POD modes the POD reduced-order solution might approximate spatial noise and artifacts which come from the adaptive finite element meshes.

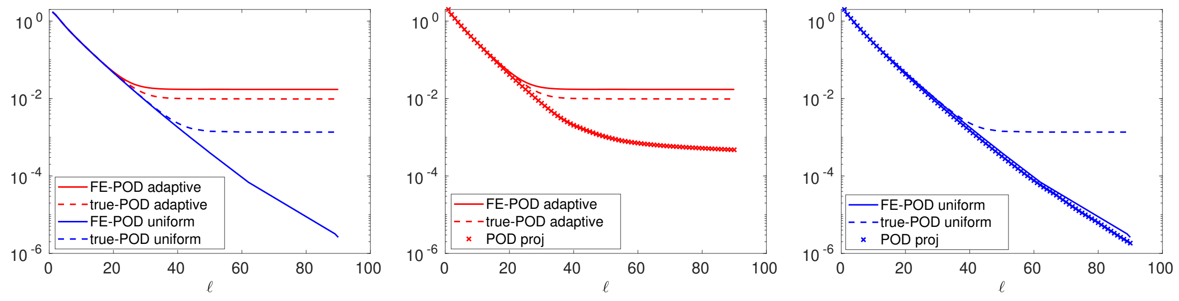

Figure 38: Relative $L^2(0,T;\Omega)$-error between the POD reduced-order solution and the finite element solution / the true solution for space-adapted and static snapshots; POD projection error. Left: errors for both static and space-adapted snapshots, middle: space-adapted snapshots, right: static snapshots

Finally, of particular interest is the computational efficiency of the POD reduced-order modeling utilizing adaptive finite element discretizations. In Table 4, the offline and online computational times are listed for the case of static and adaptive finite element snapshot generation. We observe

- For the snapshot generation we gain a speed up of factor 13.4 in the case of adaptive discretization in comparison to using a static mesh.
- The offline computational times in case of space-adapted snapshots using a construction of an overlay mesh, interpolation of the snapshots onto this mesh, setting up and solving the eigenvalue problem and then setting up the reduced-order model takes 606 seconds (10.1

minutes). In comparison, the construction of the overlay mesh and interpolation is not needed for a static finite element discretization. However, setting up the eigenvalue problem and the reduced-order model is more expensive, since it involves matrix multiplications with larger dimensions due to the larger number of degrees of freedom in the uniform setting. The offline computation time in case of static finite element discretization is 948 seconds (15.8 minutes). Thus, the use of spatially adapted snapshots leads to a speed up of factor 1.5 in the offline time for computing the POD model.

- The POD online simulation time utilizing $\ell = 20$ POD modes is very fast with only 0.02 seconds and does not depend on any spatial dimension(s). This is 52650 times faster than the adaptive finite element simulation and even 707150 times faster than the finite element simulation with static meshes. This speed up even gains greater importance, if we think of multi-query scenarios like optimal control, where multiple solutions of a dynamical system are required.

process	adaptive FE	uniform FE	speed up factor
fe simulation	1053	14143	13.4
offline times			
→ construct overlay mesh and interpolate	361	–	
→ set up and solve eigenvalue problem	73	128	1.7
→ set up ROM	172	820	4.7
POD simulation	0.02	0.02	
speed up factor	52650	707150	

Table 4: Computational times (sec) for the finite element and POD reduced simulations and offline calculations utilizing $\ell = 20$ POD modes

The last remark concerns the construction of the snapshot gramian $\mathcal{K}$ in (4.14). In this numerical example, we utilize nested and hierarchical adaptive finite element mesh structures. Thus, the interpolation of the snapshot data onto the overlay mesh does not lead to any information loss and can be implemented efficiently. Thus, this approach is favorable. In comparison, we tested an elementwise computation of $\mathcal{K}$ involving the explicit computation of inner products of pairs of snapshots which leads to the same matrix but is more involving. We exploit the nested and hierarchical structures, such that a boundary representation of the integral is not needed. Due to symmetry of $\mathcal{K}$ we only calculate the entries on and above the diagonal of $\mathcal{K}$, which are $\sum_{k=1}^{n+1} k = \frac{1}{2}((n+1)^2+n+1)$ entries. For each entry the calculation time is around 0.03 seconds, which leads to a computation time of around 60000 seconds (16.6 hours) for the matrix $\mathcal{K}$. The same effort is needed to build $\mathcal{Y}^*\mathcal{A}\mathcal{Y}$. In this case, the offline phase even takes longer than the computational times for the static case. A parallelization can be applied in order to speed up computations.

4.5.2 Cahn–Hilliard system

As a second example for the semilinear evolution problem (4.4) we consider a phase field system described by the Cahn–Hilliard equations with an additional transport term given by

$$\varphi_t + y \cdot \nabla\varphi - b\Delta\mu = 0 \quad \text{in } (0,T] \times \Omega, \tag{4.41a}$$

$$-\sigma\varepsilon\Delta\varphi + \frac{\sigma}{\varepsilon}\mathcal{F}'(\varphi) = \mu \quad \text{in } (0,T] \times \Omega. \tag{4.41b}$$

We denote by φ the phase field, μ is the chemical potential, $\mathcal{F}$ is the free energy, y is a given velocity field, b is the mobility, σ is the surface tension and ε is the interface parameter. For more details on phase field systems, we recall Section 2.5. In particular, the Cahn–Hilliard system is a nonlinear system. As choices for the free energy $\mathcal{F}$ we consider the polynomial free energy $\mathcal{F}_{pol}$ (2.22) and the relaxed double-obstacle free energy $\mathcal{F}_{s,r}$ (2.23). In order to derive a variational form of type (4.4), we write (4.41) as a single fourth-order parabolic equation for the phase field φ by eliminating μ. This leads to

$$\varphi_t + y \cdot \nabla\varphi - b\Delta(-\sigma\varepsilon\Delta\varphi + \frac{\sigma}{\varepsilon}\mathcal{F}'(\varphi)) = 0 \quad \text{in } (0,T] \times \Omega. \tag{4.42}$$

We equip the system with Neumann boundary conditions

$$\frac{\partial\varphi}{\partial\vec{n}} = 0 \quad \text{and} \quad \frac{\partial}{\partial\vec{n}}(-\sigma\varepsilon\Delta\varphi + \frac{\sigma}{\varepsilon}\mathcal{F}'(\varphi)) = 0 \quad \text{on } [0,T] \times \partial\Omega,$$

and an initial condition for the phase field

$$\varphi(0,\cdot) = \varphi_0 \quad \text{in } \Omega. \tag{4.43}$$

We choose $V = H^2(\Omega) \cap H_0^1(\Omega)$ and $H = L^2(\Omega)$ and introduce the bilinear form $a : V \times V \to \mathbb{R}$ by

$$a(\varphi, v) = \int_\Omega b\sigma\varepsilon\Delta\varphi\Delta v + (y \cdot \nabla\varphi)v dx$$

and define the nonlinear operator $\mathcal{N}$ by

$$\mathcal{N}(\varphi) = -\frac{b\sigma}{\varepsilon}\Delta\mathcal{F}'(\varphi).$$

Then, the weak form of (4.42) has the form (4.4)-(4.5).

For the actual numerical solution, we utilize the coupled form of the Cahn–Hilliard equation, which allows to use piecewise linear, continuous $\mathbb{P}_1 - \mathbb{P}_1$ finite elements, see Remark 2.48. For the time stepping we use an implicit scheme with a convex-concave splitting for the treatment of the nonlinearity. For more details we refer to the paragraph *Time discretization* in Section 2.5.2. For the solution of the nonlinear equations, we utilize Newton's method with tolerance `tol`$= 10^{-08}$ and the maximal number of iterations is `maxiter`$= 25$.

The use of h-adaptive finite elements is fundamental, since we consider a diffuse interface approach. Many degrees of freedom are required at the interfacial regions in order to well reflect the steep transitions, whereas in regions of pure phases, little numbers of degrees of freedom suffice. The use of a uniform mesh for such problems would drastically enlarge the computational effort, such that numerical simulations might be too slow for the requirements of the application or cannot be performed as often as needed in a certain amount of time. Moreover, storage problems might occur.

Problem setting

The choices for the problem data and parameters are listed in Table 5. In order to ensure time step stability of the numerical scheme, we need to impose a CFL (Courant–Friedrichs–Lewy)-type condition on each triangle $\mathcal{T}$ by

$$\max_{x\in\mathcal{T}} |y(x)| < \frac{h_T}{\Delta t}.$$

This condition means that the analytical transport of information is slower than the discrete one. The numerical method is then able to transport information with velocity y. In order to ensure an unconditionally stable system, we choose for simplicity a fixed time step size Δt which is small enough such that

$$\Delta t < \frac{h_{min}}{y_{max}},$$

where $y_{max} = 30$ and $h_{min} = 0.00625$. The choice of the time step size $\Delta t = 2.5 \cdot 10^{-05}$ fulfills this requirement.

	name	value
Ω	spatial domain	$(0, 2) \times (0, 1) \subset \mathbb{R}^2$
T	end time	0.03
b	constant mobility	$2 \cdot 10^{-05}$
ε	interface parameter	0.02
σ	surface tension	25.98
$\mathcal{F}$	nonlinear free energy	$\mathcal{F}_{pol}$ (2.22) and $\mathcal{F}_{s,r}$ (2.23) with $s = 10^4, r = 3$, respectively
y	velocity	$y = (y_1, 0)$ with $y_1(x) = \begin{cases} 30 \cdot (1 - ((x_2 - 0.5)/0.35)^2) & \text{if } \lvert x_2 - 0.5\rvert < 0.35 \\ 0 & \text{else} \end{cases}$ for $x = (x_1, x_2) \in \Omega$
φ_0	initial condition	circle with radius 0.35 and center $(0.5, 0.5)$

Table 5: Problem data and parameters of (4.41) and (4.43)

Finite element results

We start the adaptive finite element Algorithm 1 with an initial triangulation $\mathcal{T}_h^{init}$ of the spatial domain Ω which we choose as a structured criss-cross mesh with 21 node points in x_1-direction and 11 node points in x_2-direction, see Figure 40 (left). For the Dörfler marking in Definition 2.34, we set $\theta = 0.5$ and utilize a coarsening strategy according to [114, Section 8.3] with tolerance $\theta^c = 0.01$.

Figure 39 shows the finite element solution for the phase field φ and the corresponding adapted meshes at three time instances. The initial condition (a circle) is transported horizontally through the spatial domain and is deformed according to the given parabolic velocity profile. Note that the use of the polynomial free energy leads to a violation of the restriction $|\varphi| \leq 1$, whereas this condition is complied better in the case of the relaxed double-obstacle free energy. Moreover, it can be recognized that in the case of the relaxed double-obstacle free energy the phase transitions are sharper. This leads to a smaller number of necessary degrees of freedom in the adapted meshes compared to the case of the polynomial free energy. In fact, the number of degrees of freedom in the adapted meshes varies between 8235 and 15795 in the case of the relaxed double-obstacle free energy and between 10923 and 25756 in the case of the polynomial free energy.
The overlay of the adapted finite element meshes is displayed in Figure 40. It has 51168 degrees of freedom in the case of the polynomial free energy and 48653 degrees of freedom in the case of the relaxed double-obstacle free energy. In comparison, a uniform discretization with discretization fineness as small as the smallest triangle in the adaptive meshes has 102881 degrees of freedom. The computational times for the finite element simulation on adapted and uniform static spatial meshes for different free energies are listed in Table 6. For the polynomial free energies only 3 Newton steps are needed on average for each time instance, whereas for the relaxed double-obstacle free energy 5 Newton steps are needed on average. The finite element simulation time using a static uniform spatial discretization takes in the case of the polynomial free energy 30447 seconds (8.5 hours) and 100772 seconds (27.9 hours) for the relaxed double-obstacle free energy. Thus, the use of spatial adaptivity leads to an offline speed up of factor 4.5 and 8.5, respectively.

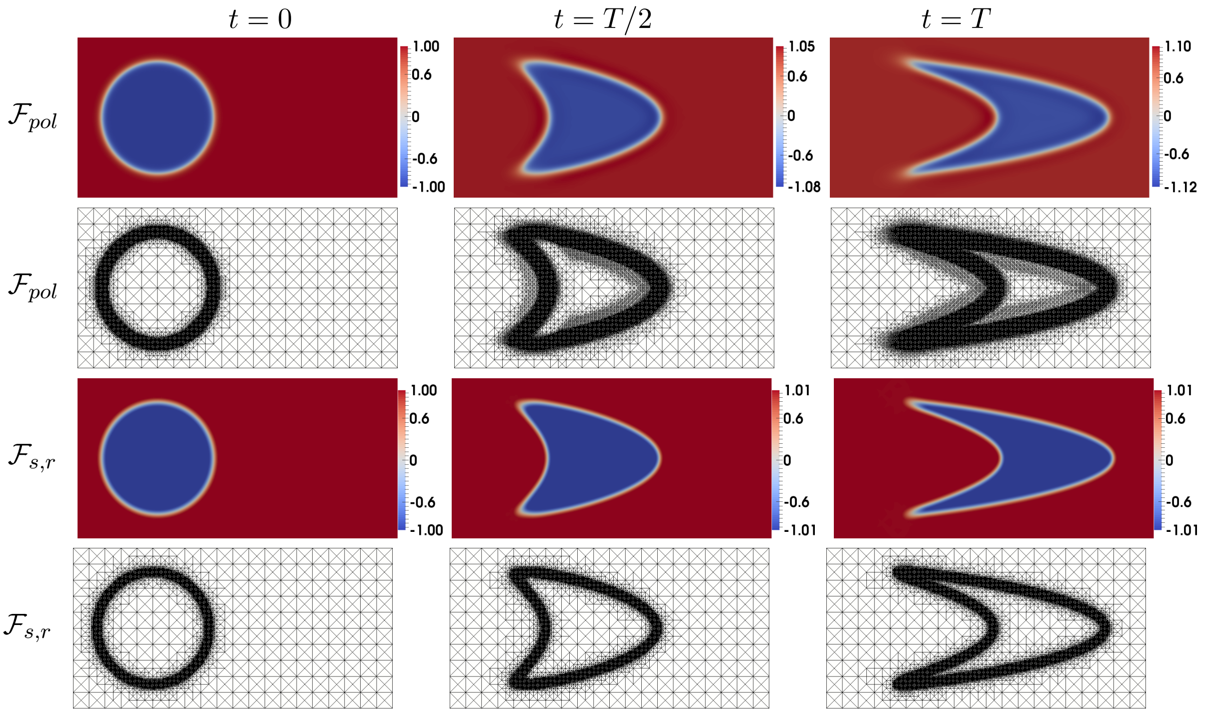

Figure 39: Finite element solution and adapted meshes for the phase field φ at three time instances comparing different free energies $\mathcal{F}$

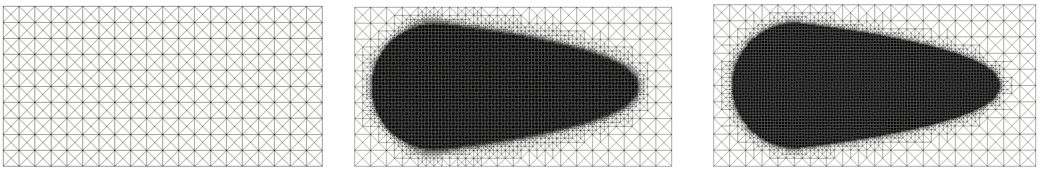

Figure 40: From left to right: macro mesh and overlay meshes for $\mathcal{F}_{pol}$ and $\mathcal{F}_{s,r}$

POD reduced-order results

Since we use nested hierarchical meshes for the adaptive finite element solution, we compute the POD basis by expressing all snapshots with respect to a common finite element basis. As snapshots we choose the finite element solutions for the phase field φ. The resulting POD basis is then used in a POD Galerkin ansatz for both phase field φ and chemical potential μ. The goal of this example is to investigate the influence of spatial adaptivity for snapshot generation in regard to the accuracy and efficiency of the reduced-order model for different free energies.

Figure 41 shows the decay of the normalized eigenspectrum for the phase field φ and the convex part $\mathcal{F}'_+$ of the nonlinearity utilizing an adaptive versus a static finite element discretization. Moreover, the use of the polynomial free energy is compared with the Moreau–Yosida relaxed double-obstacle free energy. We note for the phase field φ in the case of using the polynomial free energy that about the first 60 eigenvalues computed corresponding to space-adapted snapshots coincide with the eigenvalues corresponding to static snapshots. Then, the eigenvalues corresponding to the static discretization fall below machine precision whereas the eigenvalues for the adaptive discretization stagnate in the order of around 10^{-12}. Similar observations apply for the convex part of the nonlinearity.
In the case of the relaxed double-obstacle free energy, the decay of the eigenvalues is slower such that more POD modes are needed than in the smooth case in order to achieve the same approximation quality. This behavior is similar to the decay of the Fourier coefficients in the

context of trigonometric approximation, where the decay of the Fourier coefficients depends on the smoothness of the approximated object.

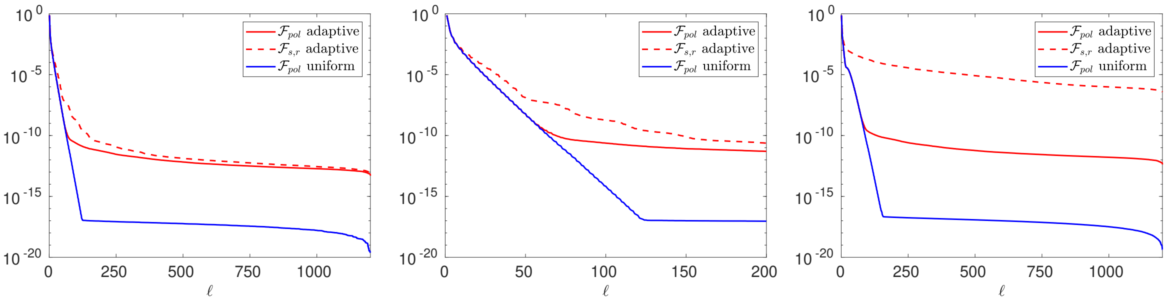

Figure 41: Decay of the normalized eigenspectrum for the phase field φ (left: all eigenvalues, middle: first 200 eigenvalues) and the convex part $\mathcal{F}'_+$ (right) for different free energies using h-adaptive and static finite element discretizations, respectively

Since the first few POD modes capture the most important information, they are shown in Figure 42 related to space-adapted snapshots. Note that the POD modes are unique up to the sign. Analogue to Section 4.5.1, we observe a periodicity in the POD basis functions corresponding to its index number.

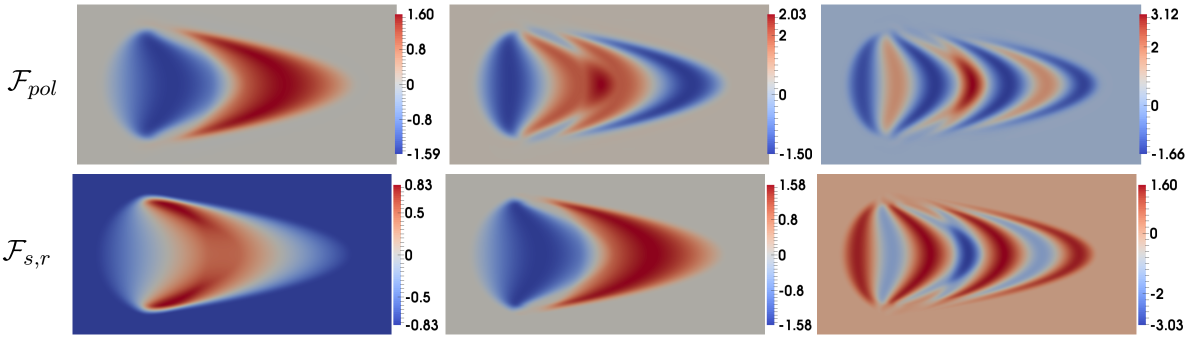

Figure 42: POD basis functions ψ_1 (left), ψ_2 (middle) and ψ_5 (right) for polynomial and relaxed double-obstacle free energy computed from space-adapted snapshots of φ

In order to detach the nonlinear POD reduced-order model from the finite element dimension, we apply the hyper-reduction technique DEIM [50] and choose the same number of DEIM basis functions as POD modes. We visualize in Figure 43 the POD reduced-order solution utilizing DEIM for the polynomial free energy and different POD basis lengths. It can be observed that the oscillations due to the transport of the phase field are smoothened out with an increasing number of POD modes.

However, we note that in the case of the relaxed double-obstacle free energy, the application of DEIM is not always successful in the sense that the POD-DEIM reduced-order model does not always converge. We observe in Figure 41 (right) that the decay of the eigenvalues for the convex part of the relaxed double-obstacle free energy is very flat. The convex part $\mathcal{F}'_+$ is zero in the interval $[-1, 1]$, including the pure phase values ± 1. For values smaller than -1 or larger than $+1$, the function $\mathcal{F}'_+$ takes very small respectively very large values (compare Figure 44). This turns out to be challenging for the application of DEIM. A normalization of the snapshots for the nonlinearity could not resolve the stability issue in our experiments. In a series of test runs, we figured out that neither increasing the exponent r to $r = 4$ or decreasing the parameter s ($s = 1000$, $s = 100$) could solve this issue. Moreover, the use of a fully implicit time discretization

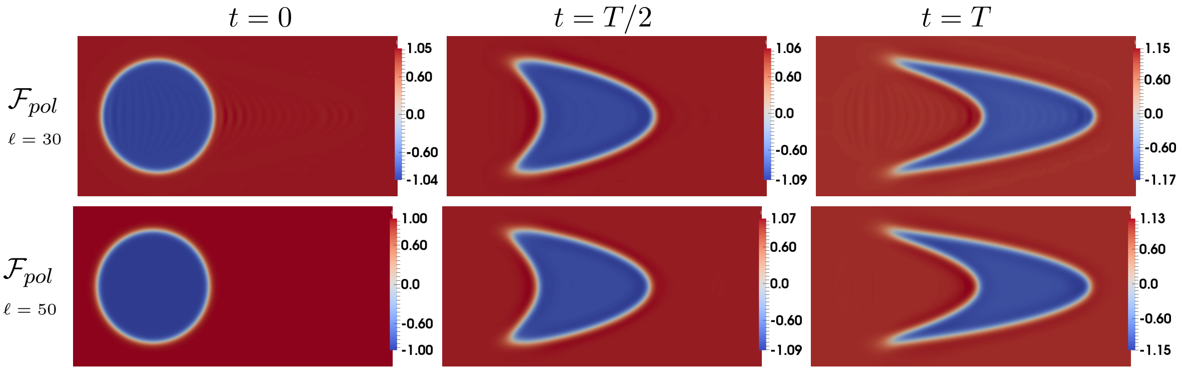

Figure 43: POD reduced-order solution for the phase field φ utilizing DEIM for $\mathcal{F}_{pol}$ and space-adapted snapshots at three time instances comparing different POD basis lengths

scheme could not alleviate the stability issues. In a numerical study in [85], we used a separate POD Galerkin ansatz for the phase field φ and the chemical potential μ. Likewise, for the resulting POD-DEIM reduced-order model we observe stability issues in the case of the relaxed double-obstacle free energy. A future research direction could be to develop a hyper-reduction method for such nonlinearities. We further note that in [182] it is proved that the POD-DEIM reduced-order model of the Allen–Cahn equation is energy stable considering the double-well and logarithmic potential and using discontinuous Galerkin discretization for the snapshots and an energy stable average vector field method for the time discretization. A transfer of these ideas to our problem setting of the Cahn–Hilliard model with a relaxed double-obstacle free energy could be of interest.

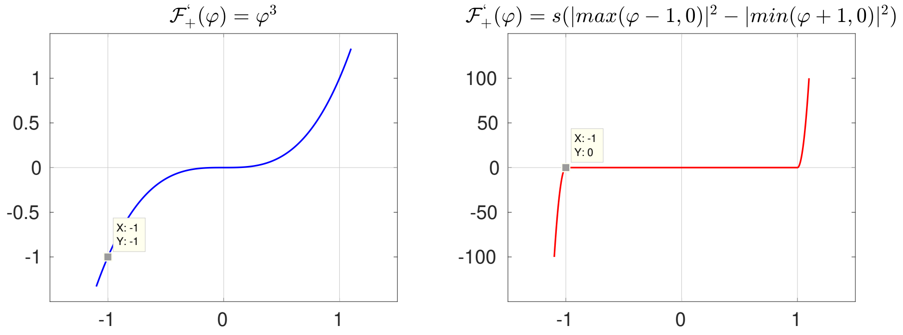

Figure 44: Convex part $\mathcal{F}_+'$ of the polynomial free energy (left) and the relaxed double-obstacle free energy (right)

The accuracy of the POD-DEIM reduced-order solution using the polynomial free energy $\mathcal{F}_{pol}$ is compared to the POD reduced-order solution using the relaxed double-obstacle free energy $\mathcal{F}_{s,r}$ in case of space-adapted snapshots in Figure 45. In particular, we consider the relative error between the finite element solution and the reduced-order solution given by

$$err_{FE-POD} = \frac{\|\varphi_h - \varphi_\ell\|_{L^2(0,T;\Omega)}}{\|\varphi_h\|_{L^2(0,T;\Omega)}}$$

As a reference, we plot the POD projection error on a uniform and adaptive mesh, respectively, which is defined as

$$err_{proj} = \frac{\|\varphi_h - \sum_{i=1}^{\ell}(\varphi_h, \psi_i)_{L^2(\Omega)}\|_{L^2(0,T;\Omega)}}{\|\varphi_h\|_{L^2(0,T;\Omega)}} = \frac{\sum_{i>\ell}\lambda_i}{\sum_{i=1}\lambda_i}.$$

In the case of a uniform static finite element discretization, we observe a convergence of the POD reduced-order solution to the finite element solution. In the case of adaptive finite element discretization for the snapshots we observe a stagnation of the relative error due to the dominance of the finite element discretization error at this level. This is similar to the results of the previous example, compare Figure 38.

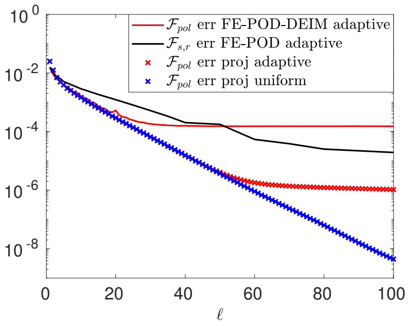

Figure 45: Relative $L^2(0,T;\Omega)$-error in φ between the reduced-order solution and the finite element solution for space-adapted snapshots for the polynomial free energy $\mathcal{F}_{pol}$ (POD-DEIM) and the relaxed double-obstacle free energy $\mathcal{F}_{s,r}$ (POD), respectively. As a reference the POD projection error is shown for adaptive and static discretizations

Table 6 summarizes the computational times for static and adaptive finite element snapshot generation with reduced-order simulation times (dimension $\ell = 30$) using the polynomial and the relaxed double-obstacle free energy, respectively. We note that POD reduced-order models lead to a speed up of 5.7 to 9.3 in comparison to the adaptive finite element simulations. Especially, the inclusion of DEIM (we use the same number of DEIM basis functions as POD modes) in the reduced-order model leads to immense speed up factors of around 30000 for both free energy functions. Note that the speed up factors are of particular interest in the context of optimal control problems. As a reference, we compare the results to a linearization of the equation (recall Section 4.3.2). The resulting relative errors between the finite element solution and the considered reduced-order solutions are provided.

process	CPU times (sec)		relative error	
	$\mathcal{F}_{pol}$	$\mathcal{F}_{s,r}$	$\mathcal{F}_{pol}$	$\mathcal{F}_{s,r}$
static finite element simulation	30447	100772		
adaptive finite element simulation	6722	11861		
interpolation onto overlay mesh	690	566		
POD basis computation	18	15		
DEIM basis computation	12	11		
POD-ROM	718	1209	$1.75 \cdot 10^{-04}$	$5.20 \cdot 10^{-04}$
POD-ROM-DEIM	0.22	0.38	$1.84 \cdot 10^{-04}$	$3.67 \cdot 10^{-03}$
POD-ROM linearized	0.07	0.07	$2.85 \cdot 10^{-03}$	$1.94 \cdot 10^{-02}$
speed up adaptive FE vs POD-ROM	9.3	5.7		
speed up adaptive FE vs POD-ROM-DEIM	30554	31213		
speed up adaptive FE vs POD-ROM linearized	96028	98971		

Table 6: Left: computational times (sec) for the finite element and reduced-order simulations and offline calculations utilizing $\ell = 30$ POD modes for different free energies. Right: relative errors between the finite element solution and the reduced-order solution as specified in the table left

4.5.3 Linear heat equation with non-nested grids

We note that this example is published in [85, Example 6.3]. Let us consider again the linear heat equation with homogeneous Dirichlet boundary data (4.40) from Section 4.5.1. The purpose of this example is to confirm the numerical applicability of the strategy described in Section 4.2.2. We set up the matrix $\mathcal{K}$ for snapshots given on non-nested meshes which requires the integration over cut elements. The data is chosen as follows: as spatial domain we choose $\Omega = (0,1) \times (0,1) \subset \mathbb{R}^2$, the time interval is $[0,T] = [0,1]$, and we utilize a uniform temporal discretization with time step size $\Delta t = 0.01$. We construct an example such that the analytical solution is known. It is given by

$$\bar{y}(t,x) = \sin(\pi x_1) \cdot \sin(\pi x_2) \cdot \cos(2\pi t x_1),$$

with $x = (x_1, x_2)$. The source term f and the initial condition g are chosen accordingly. The initial condition is projected onto a finite element space with piecewise linear and continuous node functions on a uniform spatial mesh which is shown in Figure 46 (left). Then, at each time step, the mesh is disturbed by relocating each mesh node according to the assignment

$$\begin{aligned} x_1 &\leftarrow x_1 + \theta \cdot x_1 \cdot (x_1 - 1) \cdot (\Delta t/10), \\ x_2 &\leftarrow x_2 + \theta \cdot 0.5 \cdot x_2 \cdot (x_2 - 1) \cdot (\Delta t/10), \end{aligned}$$

where $\theta \in \mathbb{R}_+$ is sufficiently small such that all coordinates of the inner node points fulfill $0 < x_1 < 1$ and $0 < x_2 < 1$. After relocating the mesh nodes, the heat equation is solved on this mesh for the next time instance. For this, we use the Lagrange interpolation in order to interpolate the finite element solution of the previous time step onto the new mesh. The disturbed meshes at $t = 0.5$ and $t = 1.0$ as well as an overlap of two meshes are shown in Figure 46. Now we have constructed finite element snapshots on non-nested meshes. For this reason, we follow the strategy explained in Section 4.2.2 and compute the matrix $\mathcal{K}$ from (4.14) by evaluating the inner products of the snapshots, which requires the integration over cut elements.

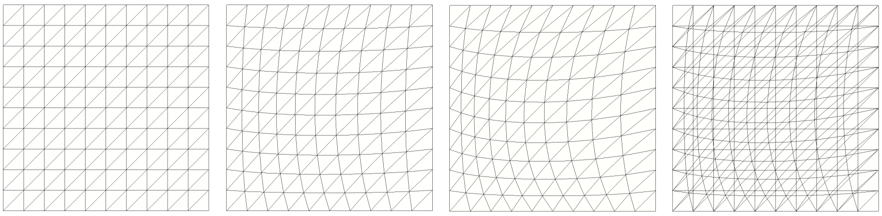

Figure 46: From left to right: Uniform mesh, disturbed meshes at $t = 0.5$ and $t = 1.0$ for $\theta = 10$, overlap of the mesh at $t = 0$ with the mesh at $t = 1.0$

We compute the eigenvalue decomposition of $\mathcal{K}$ for different values of θ and compare the results with a uniform mesh (i.e. $\theta = 0$) in Figure 47. We note that the eigenvalues of the disturbed mesh are converging to the eigenvalues of the uniform mesh for $\theta \to 0$. As expected, the eigenvalue spectrum depends only weakly on the underlying mesh given assuming that the mesh size is sufficiently small. The POD reduced-order modeling then follows along the lines of Sections 4.5.1 and 4.5.2.

Our last remark concerns the computational complexity. Solving the heat equation takes 2.1 seconds on the disturbed meshes and 1.8 seconds on the uniform mesh. The computational time for each entry of the matrix $\mathcal{K}$ is 0.022 seconds and computing the eigenvalue decomposition for $\mathcal{K}$ takes 0.0056 seconds. Note that the cut element integration problem for each matrix entry takes a fraction of time required to solve the finite element problem and can be performed in parallel.

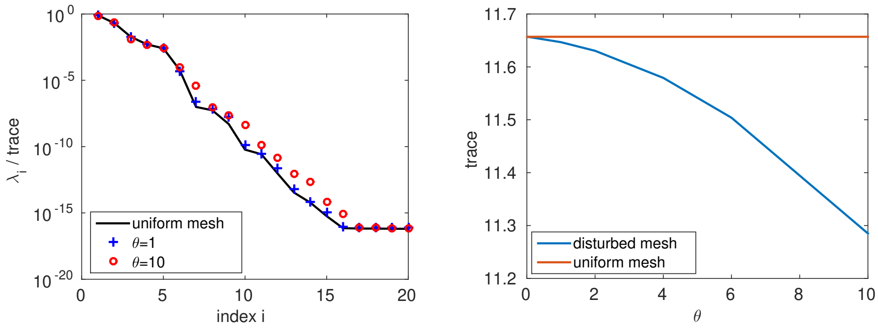

Figure 47: Decay of the eigenvalues of the matrix $\mathcal{K}$ utilizing different meshes

4.6 POD model order reduction with space-adapted snapshots for incompressible flows

This section is concerned with POD model order reduction for incompressible flows in the case of h-adapted finite element snapshots. In this case, the difficulty arises that a weak divergence-free property only holds true in the respective finite element space at the current time level and thus the challenge appears to derive a stable reduced-order model. We propose two solution concepts. The first approach is a velocity reduced-order model (Section 4.6.3) and is based on projecting the snapshots or the velocity POD basis onto a reference velocity space. This projection is constructed such that the resulting velocity POD modes are weakly divergence-free with respect to a reference pressure space. The second approach is a velocity-pressure reduced-order model (Section 4.6.4) and guarantees stability by enriching the velocity POD space with pressure supremizer functions. In this way, it can be shown that the inf-sup stability condition from the finite element model carries over to an inf-sup stability condition for the reduced-order model. Moreover, we address inhomogeneous Dirichlet data by homogenization and constructing a certain lifting function (Section 4.6.5). As a numerical example, we present in Section 4.6.6 the benchmark problem of a lid-driven cavity flow utilizing h-adapted finite element snapshots. A comparison of the two proposed solution approaches is presented with regards to efficiency and accuracy.

We note that this section is based on a collaboration with Michael Hinze, Jens Lang and Sebastian Ullmann. A resulting manuscript [86] is accepted for publication in Advances in Computational Mathematics in 2019.

Literature overview

Let us give an overview of publications which are related to model order reduction of incompressible flows or address stability issues. Due to the amount of scientific contributions to this issue, we only give a selection of publications here and also refer to the references therein.
The essential idea to identify coherent structures of spatial velocity correlations is given in [133]. In the pioneer work [166, 167, 168], the method of snapshots is introduced. The incompressibility of the POD modes in case of a static discretization is then used in order to derive a reduced-order model which only depends on the velocity and the pressure term vanishes. Different problem settings of a channel flow and a Boussinesq approximation are investigated.
The publication [29] provides a review on proper orthogonal decomposition in the analysis of turbulent flows with particular attention to the question what can be learnt about the original partial differential equation.
In [111] reduced-order modeling for simulation and control of fluid flows is presented. Two numerical experiments of a velocity tracking problem in cavity flow and a vorticity control problem in a channel flow demonstrate the efficiency and feasibility of the reduced-order approximation.

Stability properties of POD reduced-order approximations for a compressible Navier–Stokes system are the content of [110]. POD is interpreted as a filtering process. Moreover, it is suggested to utilize the H^1-norm in the POD formulation.
In [121], POD model order reduction is applied to a general equation in fluid dynamics including the case of the Navier–Stokes equations in two dimensions. The convergence of the POD surrogate model is analyzed. The analysis comprises the case in which different time discretization schemes are used for the full-order and the reduced-order simulations. Perturbation analysis is carried out for $\Delta t \to 0$.
In [184], a parametrized steady incompressible Navier–Stokes system is considered. The reduced basis method it utilized in order to provide real-time and certified approximations for deployed systems. Rigorous reduced basis a-posteriori error bounds are provided.
In [41], POD and CVT (centroidal Voronoi tessellation) reduced-order modeling of the Navier–Stokes equations in 2d are compared. For the numerical example of a T-cell problem, both approaches seem to be equally effective.
A static unstructured mesh modeling is utilized in [62] in order to resolve both small- and large-scale fluid flows in 2d and 3d settings. Starting from a uniform mesh, adaptive mesh refinement is utilized in order to achieve a relatively fine mesh. This mesh is then fixed and the POD method is carried out in the usual manner.
For a general review of challenges an perspectives in model order reduction for fluid dynamic settings, we refer to [128]. General principles of reduced-order modeling for parametrized are discussed, applications to steady and unsteady flow problems are provided and the question of inf-sup stability is addressed.
Of particular importance for the following is the supremizer stabilization proposed in [160, 22]. A parametrized setting of the Stokes and the steady incompressible Navier–Stokes equations are considered, respectively. Stability of the reduced-order models is guaranteed by enriching the reduced basis velocity space with supremizer functions. The stability is first analyzed from a theoretical point of view and then confirmed in numerical test examples.

4.6.1 Problem setting

Let $\Omega \subset \mathbb{R}^2$ be a bounded domain with Lipschitz boundary $\partial\Omega$, recall Definition 2.17, and let $T > 0$ be a fixed end time. We consider an unsteady incompressible flow problem governed by the Navier–Stokes equations which is given in strong form as follows: find a velocity field $y = (y_1, y_2)$ and a pressure p which fulfill

$$y_t + (y \cdot \nabla)y - Re^{-1}\Delta y + \nabla p = f \quad \text{in } (0,T] \times \Omega, \tag{4.44a}$$
$$\nabla \cdot y = 0 \quad \text{in } [0,T] \times \Omega, \tag{4.44b}$$
$$y = 0 \quad \text{on } [0,T] \times \partial\Omega, \tag{4.44c}$$
$$y(0,\cdot) = y_0 \quad \text{in } \Omega. \tag{4.44d}$$

The equation (4.44a) is called *momentum equation*, the (strong) divergence-free constraint (4.44b) is referred to as *continuity equation*. By Re we denote the Reynolds number, f is a given body force and y_0 denotes the initial velocity field. Note that we consider a consistent setting in the sense that the initial data fulfills the boundary condition (4.44c) and is solenoidal, i.e. it fulfills the strong divergence-free property (4.44b). In the following, we have applications in mind of laminar flow with low or moderate Reynolds numbers. For a general introduction to the derivation, mathematical analysis and numerical treatment of the Navier–Stokes equations we refer to [56, 112, 173], for example.

We note that the divergence constraint (4.44b) encodes the incompressibility of the fluid, or equivalently conservation of mass. The case of inhomogeneous Dirichlet and initial data is investigated in Section 4.6.5.

Weak form

For the ease of presentation, we use the same notation for scalar- and vector-valued functions, meaning that all components of a vector-valued function belong to the corresponding scalar function space. The same holds for scalar- and vector-valued operators. We introduce the space of integrable solenoidal functions by

$$H_{div}(\Omega) := \{v \in L^2(\Omega) : \nabla \cdot v = 0, v \cdot \vec{n}|_{\partial\Omega} = 0 \text{ in the sense of traces}\}. \tag{4.45}$$

The pressure is determined only up to an additive constant such that an additional condition is introduced, e.g. by imposing the integral mean value to vanish. For this reason we introduce the space

$$L_0^2(\Omega) = \{q \in L^2(\Omega) : \int_\Omega q dx = 0\}.$$

Moreover, we define the space

$$W_*^1(0,T;H_0^1(\Omega)) = \{v \in L^2(0,T;H_0^1(\Omega)) : v_t \in L^1(0,T;H^{-1}(\Omega))\}.$$

The weak form of (4.44) reads as follows: for a given body force $f \in L^2(0,T;H^{-1}(\Omega))$ and initial value $y_0 \in H_{div}(\Omega)$ find a velocity $y \in W_*^1(0,T;H_0^1(\Omega))$ with $y(0,\cdot) = y_0$ and a pressure $p \in L^2(0,T;L_0^2(\Omega))$ such that

$$\frac{d}{dt}(y(t),v)_{L^2(\Omega)} + c(y(t),y(t),v) + a(y(t),v) + b(v,p(t)) = \langle f(t),v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} \quad \forall v \in H_0^1(\Omega), \tag{4.46a}$$

$$b(y(t),q) = 0 \quad \forall q \in L_0^2(\Omega), \tag{4.46b}$$

for almost all $t \in [0,T]$. We use the notations $a(u,v) = Re^{-1}(\nabla u, \nabla v)_{L^2(\Omega)}$, $c(w,u,v) = ((w \cdot \nabla)u,v)_{L^2(\Omega)}$ and $b(v,q) = -(q,\nabla \cdot v)_{L^2(\Omega)}$. For existence and uniqueness of a solution to (4.46) we refer to e.g. [173, Chapter 3, Theorems 3.1 and 3.2].
In order to solve (4.46) numerically, we first discretize in time and then discretize in space which allows to use a different finite element space at each time level.

Time discretization

For the temporal discretization of (4.46), we apply a fully implicit Euler scheme. Of course, other time integration schemes are possible. For a given $n \in \mathbb{N}$, we introduce a time grid $0 = t_0 < t_1 < \cdots < t_n = T$ and time step sizes $\Delta t_j = t_j - t_{j-1}$ for $j = 1,\ldots,n$. The time-discrete system reads as follows: for a given initial velocity $y_0 \in H_{div}(\Omega)$ find sequences $y^1,\ldots,y^n \in H_0^1(\Omega)$ and $p^1,\ldots,p^n \in L_0^2(\Omega)$ such that

$$\left(\frac{y^j - y^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} + c(y^j,y^j,v) + a(y^j,v) + b(v,p^j) = \langle f(t_j),v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} \quad \forall v \in H_0^1(\Omega), \tag{4.47a}$$

$$b(y^j,q) = 0 \quad \forall q \in L_0^2(\Omega). \tag{4.47b}$$

Note that we applied the box rule in order to approximate the right-hand side time integral in (4.47a). An initial pressure field can be obtained from an additional pressure Poisson equation, if required, see e.g. [94, Proposition 2.1, Corollary 2.1].

Fully discrete form

For the spatial discretization, we utilize h-adaptive finite elements based on LBB (Ladyzhenskaya–Babuška–Brezzi [19, 40, 124]) stable $\mathbb{P}_2 - \mathbb{P}_1$ Taylor–Hood element pairs denoted by (V_h^j, Q_h^j) for $j = 1, \dots, n$. Details on the construction of the adapted finite element spaces are given below. Let V_h^0 denote the finite element space associated with the initial mesh $\mathcal{T}_h^{init}$. For a given initial condition $(y_h^0, v)_{L^2(\Omega)} = (y_0, v)_{L^2(\Omega)}$ for all $v \in V_h^0$ with $y_0 \in H_{div}(\Omega)$, let us consider the fully discrete Navier–Stokes system which reads as follows: find $y_h^1 \in V_h^1, \dots, y_h^n \in V_h^n$ and $p_h^1 \in Q_h^1, \dots, p_h^n \in Q_h^n$ such that

$$\left(\frac{y_h^j - y_h^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} + c(y_h^j, y_h^j, v) + a(y_h^j, v) + b(v, p_h^j) = \langle f(t_j), v\rangle_{H^{-1}(\Omega), H_0^1(\Omega)} \quad \forall v \in V_h^j, \tag{4.48a}$$

$$b(y_h^j, q) = 0 \qquad \forall q \in Q_h^j. \tag{4.48b}$$

For existence of a unique solution to (4.48), we refer to [173, Chapter 3, §5 Scheme 5.1']. Note that in each time step the computation of the inner product of a velocity from the previous time step $y_h^{j-1} \in V_h^{j-1}$ with a test function $v \in V_h^j$ is required. In [32] it is shown that for spatially adapted mesh discretizations, the pressure diverges with order Δt^{-1} for $\Delta t \to 0$. It is shown that the discrete pressure is bounded if the L^2-projection of the velocity from the previous time instance is weakly divergence-free with respect to test functions of the current finite element velocity space. Suitable projections are proposed to circumvent the divergence of the pressure. Alternatively, in [14, Section 6.2], an index-1 formulation is used in order to derive a robust scheme.

For the construction of the adapted finite element spaces (V_h^j, Q_h^j), we follow the usual adaptive cycle, recall Figure 3. In particular, we make the following choices in Algorithm 1: for the *solution* of the nonlinear algebraic saddle point problem (4.48), we utilize Newton's method. The *estimation* of the spatial error on each triangle is based on a residual a-posteriori error estimator in the spirit of [4]. In particular, we obtain on each triangle $\mathcal{T}$ of the current triangulation $\mathcal{T}_h^j$ the error indicator

$$\eta_{\mathcal{T}}^j = \Bigg(h_{\mathcal{T}}^2 \left\| \frac{y_h^j - y_h^{j-1}}{\Delta t_j} + y_h^j \cdot \nabla y_h^j - Re^{-1} \Delta y_h^j + \nabla p_h^j - f(t_j) \right\|_{L^2(\mathcal{T})}^2 + \left\| \nabla \cdot y_h^j \right\|_{L^2(\mathcal{T})}^2 + \frac{1}{2} \sum_{\mathcal{E} \in \partial\mathcal{T} \setminus \partial\Omega} h_{\mathcal{E}} \left\| [- Re^{-1} \nabla y_h^j \cdot \vec{n} + p_h^j \vec{n}]_{\mathcal{E}} \right\|_{L^2(\mathcal{E})}^2 \Bigg)^{\frac{1}{2}} \quad \forall \mathcal{T} \in \mathcal{T}_h^j \tag{4.49}$$

for $j = 1, \dots, n$ which assumes $f(t_j) \in L^2(\Omega)$. We denote by $h_{\mathcal{T}}^2$ the triangle area and $h_{\mathcal{E}}$ is the edge length. Since we do not expect drastic dynamical changes from one time step to another for small time instances, we mark every triangle of the current triangulation and perform one coarsening step. This ensures the possibility of reaching the initial mesh $\mathcal{T}_h^{init}$ after a finite number of time steps. Although this choice might lead to a finer triangulation than actually postulated, we expect the advantage that only a small number of refinement steps are needed in the next time iteration.

4.6.2 POD reduced-order modeling

Assume we are given a reduced velocity space V_ℓ and a reduced pressure space Q_ℓ. We discuss the detailed construction later. In order to derive a reduced-order model, we replace the test spaces in (4.47) by the reduced spaces V_ℓ and Q_ℓ, respectively. The reduced-order problem reads as follows: for given initial condition $(y_\ell^0, \psi)_{L^2(\Omega)} = (y_0, \psi)_{L^2(\Omega)}$ with $y_0 \in H_{div}(\Omega)$ find $y_\ell^1, \dots, y_\ell^n \in V_\ell$ and $p_\ell^1, \dots, p_\ell^n \in Q_\ell$ such that

$$\left(\frac{y_\ell^j - y_\ell^{j-1}}{\Delta t_j}, v \right)_{L^2(\Omega)} + c(y_\ell^j, y_\ell^j, v) + a(y_\ell^j, v) + b(v, p_\ell^j) = \langle f(t_j), v \rangle_{H^{-1}(\Omega), H_0^1(\Omega)} \quad \forall v \in V_\ell, \tag{4.50a}$$

$$b(y_\ell^j, q) = 0 \qquad \forall q \in Q_\ell. \tag{4.50b}$$

This is a system of algebraic equations for the expansion coefficients of the reduced solutions and does not depend on the finite element dimension, see [168, Section 1]. Note that the stability of the system (4.50) is not guaranteed for all pairs (V_ℓ, Q_ℓ). In the following we propose two choices for (V_ℓ, Q_ℓ) which lead to stable reduced-order models.

The first approach (Section 4.6.3) is a velocity reduced-order model. It relies on a reference pressure space paired with a velocity POD space, where the POD basis functions are weakly divergence-free with respect to the reference pressure space. This enables a cancellation of the pressure term in (4.50a) and the continuity equation (4.50b) is fulfilled by construction. The stability of the resulting system is then given by [56, 173].

The second approach (Section 4.6.4) is a velocity-pressure reduced-order model. Separate POD basis functions are computed for the velocity and for the pressure. Stability is achieved by enriching the velocity POD space with pressure supremizer functions in the spirit of [22, 160].

In the following we make use of a reference velocity space $\tilde{V} \subset H_0^1(\Omega)$ and a corresponding reference pressure space $\tilde{Q} \subset L_0^2(\Omega)$, such that the pair $(\tilde{V}, \tilde{Q})$ is inf-sup stable. These spaces do not have to fulfill further assumptions and are therefore kept general. One option could be to choose common finite element spaces $\tilde{V}$ and $\tilde{Q}$ which contain all finite element spaces, i.e. $V_h^1, \dots, V_h^n \subset \tilde{V}$ and $Q_h^1, \dots, Q_h^n \subset \tilde{Q}$. For example, if the meshes $\mathcal{T}_h^1, \dots, \mathcal{T}_h^n$ are nested, hierarchical meshes, then the spaces $\tilde{V}$ and $\tilde{Q}$ can be constructed associated with the overlay of all meshes. However, it is also possible to choose $\tilde{V}$ and $\tilde{Q}$ independently of the snapshot spaces. This enables to decouple the reduced-order solution from the finite element discretization.

4.6.3 Velocity reduced-order model

In this section, we derive a POD reduced-order model which only depends on the velocity variable while the pressure term and the continuity equation are canceled out. This approach is in the spirit of [166, 167] and based on the fact that the POD modes are weakly divergence-free. The challenge in the context of spatially adapted snapshots is that *weakly divergence-free* refers to the test space which can be a different space at each time instance. More precisely, for $j = 1, \dots, n$ the solution $y_h^j \in V_h^j$ of (4.48) fulfills a weak divergence-free property with respect to the corresponding pressure space Q_h^j, i.e.

$$b(y_h^j, q) = 0 \quad \forall q \in Q_h^j.$$

However, in general y_h^j is not weakly divergence-free with respect to the other pressure spaces Q_h^i for $i \neq j$. Thus, for a linear combination of the snapshots it cannot be guaranteed that a weak divergence-free property is fulfilled in a common sense. For this reason, the usual velocity ROM does not work since the POD basis functions are a linear combination of the snapshots.

In order to tackle this issue, we construct a *modified* velocity POD basis which is weakly divergence-free with respect to a reference pressure space. Two approaches of constructing a suitable modified velocity POD basis are proposed. The first strategy is based on a projection of the original snapshots meaning that we first project the snapshots such that they are weakly divergence-free in a common sense and then compute a POD basis from the projected snapshots (*first-project-then-reduce*). The second approach is based on projected POD modes, meaning that we first compute a POD basis from the original snapshots and then project the POD basis functions such that they are weakly divergence-free in a common sense (*first-reduce-then-project*).

Optimal projection onto a weakly divergence-free space

Our aim is to define weakly divergence-free approximations in a more general sense. We provide a procedure that can be applied to problems with inhomogeneous Dirichlet data as well. To this end, we introduce a Dirichlet lifting function g, which will be specified in Section 4.6.5. In the context of homogeneous Dirichlet data, we set $g \equiv 0$.

For a given function u in a Hilbert space X we want to find a function $\tilde{u} \in \tilde{V} \subset X$ such that $\tilde{u} + g$ is weakly divergence-free with respect to a space $\tilde{Q}$. To be more precise, we want to solve the following equality constrained minimization problem:

Problem 4.5. *For given $u \in X$ and sufficiently smooth g, find $\tilde{u} \in \tilde{V} \subset X$ which solves*

$$\min_{v \in \tilde{V}} \frac{1}{2} \|v - u\|_X^2 \quad \textit{subject to} \quad b(v + g, q) = 0 \quad \forall q \in \tilde{Q}.$$

Note that Problem 4.5 has a unique solution $\tilde{u} \in \tilde{V}$. We define the projection $u \mapsto P_g(u) = \tilde{u}$. In order to compute the solution to Problem 4.5, the usual Lagrange approach can be followed, recall Section 2.3. The resulting optimality system is given by the following saddle point problem:

Problem 4.6. *For given $u \in X$ and sufficiently smooth g, find $\tilde{u} \in \tilde{V} \subset X$ and $\lambda \in \tilde{Q}$ such that*

$$\begin{aligned} (\tilde{u}, w)_X + b(w, \lambda) &= (u, w)_X && \forall w \in \tilde{V}, \\ b(\tilde{u}, q) &= -b(g, q) && \forall q \in \tilde{Q}. \end{aligned}$$

Note that the solution $\tilde{u} \in \tilde{V}$ to Problem 4.6 is unique. If $(\tilde{V}, \tilde{Q})$ forms an inf-sup stable pair, then $\lambda \in \tilde{Q}$ is unique. For the choice $X = H_0^1(\Omega)$ the Problem 4.6 is a Stokes problem, for $X = L^2(\Omega)$, the Problem 4.6 is a weak form of the Leray projection.

a) *First-project-then-reduce*

The basic idea of this approach is to project the original snapshots, i.e. the velocity solution of (4.48), in order to obtain functions which are weakly divergence-free with respect to the reference pressure space $\tilde{Q}$. As a consequence, the resulting POD basis inherits the property of weak-divergence by construction. In particular, for given snapshots $y_h^1 \in V_h^1, \dots, y_h^n \in V_h^n$ we solve Problem 4.5 with $g \equiv 0$. Then, the projected snapshots $\tilde{y}^1 = P_0(y_h^1), \dots, \tilde{y}^n = P_0(y_h^n)$ live in

$$\tilde{V}_{div} := \{v \in \tilde{V} : b(v, q) = 0 \quad \forall q \in \tilde{Q}\}.$$

From these projected snapshots $\tilde{y}^1, \dots, \tilde{y}^n$, we compute a POD basis according to (2.7) with basis length ℓ_y. The resulting POD space $V_\ell := \operatorname{span}(\psi_1, \dots, \psi_{\ell_y})$ fulfills the property $V_\ell \subset \tilde{V}_{div} \subset \tilde{V}$. Thus, $b(v, q) = 0$ holds true for all $v \in V_\ell$ and all $q \in \tilde{Q}$. Consequently, for this choice of V_ℓ in (4.50), together with $Q_\ell = \tilde{Q}$, the pressure term vanishes and the continuity equation is fulfilled by construction. The resulting velocity reduced-order model reads as follows: for a given initial condition $(y_\ell^0, \psi)_{L^2(\Omega)} = (y_0, \psi)_{L^2(\Omega)}$ with $y_0 \in H_{div}(\Omega)$, find $y_\ell^1, \dots, y_\ell^n \in V_\ell$ such that

$$\left(\frac{y_\ell^j - y_\ell^{j-1}}{\Delta t^j}, v \right)_{L^2(\Omega)} + c(y_\ell^j, y_\ell^j, v) + a(y_\ell^j, v) = \langle f(t_j), v \rangle_{H^{-1}(\Omega), H_0^1(\Omega)} \quad \forall v \in V_\ell. \tag{4.51}$$

Obviously, the velocity reduced-order model (4.51) only depends on the velocity variable and the pressure is eliminated. However, in the case of snapshot generation on static spatial meshes, it is possible to reconstruct a reduced pressure afterwards by solving a discrete reduced pressure Poisson equation, see [180], for example. A transfer of this concept to the case of space-adapted snapshots is not carried out within the scope of this work.

Computational complexity

Concerning the computational complexity, we note that Problem 4.5 has to be solved for each snapshot, which means the solution of a saddle point problem with reference spaces $\tilde{V}$ and $\tilde{Q}$, followed by the computation of a POD basis for the velocity field. Concerning the online computational costs, in each time step we solve a nonlinear algebraic system of equations with Newton's method. In each Newton step, we need to build a Jacobian matrix and a right-hand side and, subsequently, solve a dense linear system using a direct method. Therefore, we find that solving the velocity reduced-order model (4.51) is of order $\mathcal{O}(\ell_y^3)$.

b) *First-reduce-then-project*

The basic idea of this approach is to compute a POD basis from the original velocity solutions of (4.48) and project the resulting POD basis functions afterwards such that they fulfill a weak divergence-free property. In particular, for given snapshots $y_h^1 \in V_h^1, \dots, y_h^n \in V_h^n$, we define interpolated snapshots $\hat{y}^j = \pi_{\tilde{V}} y_h^j \in \tilde{V}$ for $j = 1, \dots, n$ where $\pi_{\tilde{V}}$ denotes the Lagrange interpolation operator onto the velocity reference space $\tilde{V}$. From these interpolated snapshots, we compute POD basis functions $\hat{\psi}_1, \dots, \hat{\psi}_{\ell_y} \in \tilde{V}$ according to (2.7) with POD basis length ℓ_y. These POD basis functions in general do not live in $\tilde{V}_{div}$. Thus, they are projected onto the space $\tilde{V}_{div}$ by solving (4.5) with $g \equiv 0$. Then, the projected POD basis functions $\psi_1 = P_0(\hat{\psi}_1), \dots, \psi_{\ell_y} = P_0(\hat{\psi}_{\ell_y})$ live in $\tilde{V}_{div}$. Choosing $V_\ell := \text{span}(\psi_1, \dots, \psi_{\ell_y})$ in (4.50) together with $Q_\ell = \tilde{Q}$ leads to a velocity reduced-order model of the form (4.51). Note that, in general, the reduced space constructed in this approach does not coincide with the reduced space constructed according to *a) first-project-then-reduce*.

Computational complexity

The computational complexity of the approach described in this subsection comprises the computation of a POD basis and, afterwards, the solution of Problem 4.5 for each POD basis function, i.e. ℓ_y times. This makes the approach *first-reduce-then-project* cheaper than the approach *first-project-then-reduce*, which requires n solutions of Problem 4.5, and $\ell_y \leq n$. Then, the costs of setting up and solving the reduced-order model are equivalent.

4.6.4 Velocity-pressure reduced-order model

In this section, we derive a POD reduced-order model which depends on both velocity and pressure variable. This approach is in the spirit of [22, 160] and is based on a construction of an inf-sup stable pair of velocity and pressure reduced spaces. In particular, we compute separate POD reduced spaces for the velocity and the pressure and enrich the velocity POD space with suitable pressure supremizer functions in order to guarantee an inf-sup stability condition.

Construction of inf-sup stable POD spaces

From the velocity solutions $y_h^1 \in V_h^1, \dots, y_h^n \in V_h^n$ and pressure solutions $p_h^1 \in Q_h^1, \dots, p_h^n \in Q_h^n$ of problem (4.48), we define interpolated velocity snapshots $\hat{y}^j = \pi_{\tilde{V}} y_h^j \in \tilde{V}$ and interpolated pressure snapshots $\hat{p}^j = \pi_{\tilde{Q}} p_h^j \in \tilde{Q}$ for $j = 1, \dots, n$, where $\pi_{\tilde{V}}$ and $\pi_{\tilde{Q}}$ are the Lagrange interpolation operators onto the reference velocity space $\tilde{V}$ and the reference pressure space $\tilde{Q}$, respectively. For the velocity POD, we compute ℓ_y POD basis functions according to (2.5). As a result, we obtain velocity POD basis functions $\psi_1, \dots, \psi_{\ell_y} \in \operatorname{span}(\hat{y}^1, \dots, \hat{y}^n)$. For the pressure, we compute ℓ_p pressure POD basis functions according to (2.5). Thus, we obtain pressure POD basis functions $\phi_1, \dots, \phi_{\ell_p} \in \operatorname{span}(\hat{p}^1, \dots, \hat{p}^n)$.

The goal is to construct a pair of reduced spaces (V_ℓ, Q_ℓ) which ensures stability of the POD reduced-order model (4.50). We introduce an inf-sup stability criterion as follows: there exists a constant β_ℓ defined by

$$\beta_\ell := \inf_{\substack{q \in Q_\ell \\ q \neq 0}} \sup_{\substack{v \in V_\ell \\ v \neq 0}} \frac{b(v,q)}{\|v\|_{H_0^1(\Omega)} \|q\|_{L^2(\Omega)}}. \tag{4.52}$$

which needs to be greater than zero ($\beta_\ell > 0$). We make the following choices. As reduces spaces we take

$$\begin{aligned} V_\ell &:= \operatorname{span}(\psi_1, \dots, \psi_{\ell_y}, \bar{\psi}_1, \dots, \bar{\psi}_{\ell_p}), && \text{(4.53a)} \\ Q_\ell &:= \operatorname{span}(\phi_1, \dots, \phi_{\ell_p}), && \text{(4.53b)} \end{aligned}$$

with supremizer functions $\bar{\psi}_1, \dots, \bar{\psi}_{\ell_p}$. In the following, we explain the computation of the supremizer functions and show that this choice of reduced spaces (V_ℓ, Q_ℓ) leads to a POD reduced-order model which fulfills the inf-sup stability condition in the sense of (4.52).

a) Supremizer computation

We introduce a linear map $\mathbb{T} : L_0^2(\Omega) \to \tilde{V}$ as follows: for given $q \in L_0^2(\Omega)$, find $\mathbb{T}q \in \tilde{V}$ such that

$$(\mathbb{T}q, v)_{H_0^1(\Omega)} = b(v,q) \quad \forall v \in \tilde{V}. \tag{4.54}$$

From the Riesz representation Theorem 2.8 follows

$$\|\mathbb{T}q\|_{\tilde{V}} = \sup_{\substack{v \in \tilde{V} \\ v \neq 0}} \frac{b(v,q)}{\|v\|_{H_0^1(\Omega)}} = \sup_{\substack{v \in \tilde{V} \\ v \neq 0}} \frac{(\mathbb{T}q, v)_{H_0^1(\Omega)}}{\|v\|_{H_0^1(\Omega)}} \quad \forall q \in L_0^2(\Omega), \tag{4.55}$$

which means that

$$\mathbb{T}q = \arg \sup_{\substack{v \in \tilde{V} \\ v \neq 0}} \frac{(\mathbb{T}q, v)_{H_0^1(\Omega)}}{\|v\|_{H_0^1(\Omega)}} \quad \forall q \in L_0^2(\Omega). \tag{4.56}$$

Now, for each pressure POD function ϕ_j for $j = 1, \dots, \ell_p$ we compute a function $\bar{\psi}_j \in \tilde{V}$ which solves (4.54), i.e. $\bar{\psi}_j = \mathbb{T}\phi_j$.

b) Inf-sup stability of the velocity-pressure reduced-order model

In order to show that the choice of reduced spaces (V_ℓ, Q_ℓ) according to (4.53) is inf-sup stable in the sense of (4.52), we relate the stability constant β_ℓ of the reduced-order model with the stability constant $\tilde{\beta}_h$ of the reference spaces $(\tilde{V}, \tilde{Q})$ and proceed analogously to [22, Proposition 2] and [160, Lemma 3.1]:

$$\begin{aligned}
0 < \tilde{\beta}_h &= \inf_{\substack{q\in\tilde{Q}\\ q\neq 0}} \sup_{\substack{v\in\tilde{V}\\ v\neq 0}} \frac{b(v,q)}{\|v\|_{H_0^1(\Omega)}\|q\|_{L^2(\Omega)}} && \text{FE stability}\\
&\leq \inf_{\substack{q\in Q_\ell\\ q\neq 0}} \sup_{\substack{v\in\tilde{V}\\ v\neq 0}} \frac{b(v,q)}{\|v\|_{H_0^1(\Omega)}\|q\|_{L^2(\Omega)}} && Q_\ell \subset \tilde{Q}\\
&= \inf_{\substack{q\in Q_\ell\\ q\neq 0}} \frac{b(\mathbb{T}q,q)}{\|\mathbb{T}q\|_{H_0^1(\Omega)}\|q\|_{L^2(\Omega)}} && (4.55), (4.56)\\
&\leq \inf_{\substack{q\in Q_\ell\\ q\neq 0}} \sup_{\substack{v\in\{\mathbb{T}p:p\in Q_\ell\}\\ v\neq 0}} \frac{b(v,q)}{\|v\|_{H_0^1(\Omega)}\|q\|_{L^2(\Omega)}}\\
&\leq \inf_{\substack{q\in Q_\ell\\ q\neq 0}} \sup_{\substack{v\in V_\ell\\ v\neq 0}} \frac{b(v,q)}{\|v\|_{H_0^1(\Omega)}\|q\|_{L^2(\Omega)}} = \beta_\ell. && V_\ell = \operatorname{span}\{\psi_1,\dots,\psi_{\ell_y},\bar{\psi}_1,\dots,\bar{\psi}_{\ell_p}\}
\end{aligned}$$

Consequently, the inf-sup stability of the finite element model carries over to an inf-sup stability of the reduced-order model.

Computational complexity

Concerning the complexity of the construction of the velocity-pressure reduced-order model, there are two major differences to the velocity reduced-order model. First of all, the velocity-pressure reduced-order model requires an additional computation of pressure POD basis functions, which is not needed for the pure velocity reduced-order model. Second, the construction of the divergence-free POD modes in the velocity reduced-order approach requires the solution of Problem 4.6, whose complexity depends on $\dim(\tilde{V}) + \dim(\tilde{Q})$. In contrast, the construction of the supremizer functions in the velocity-pressure reduced-order approach requires the solution of (4.54), whose complexity depends on $\dim(\tilde{V})$. Concerning the online computational costs, the solution of the velocity-pressure reduced-order model is of order $\mathcal{O}((\ell_y + \ell_p)^3)$.

4.6.5 Inhomogeneous Dirichlet and initial data

So far, we considered the incompressible Navier–Stokes system with homogeneous Dirichlet data. In the following, we extend the scope to problems involving inhomogeneous boundary conditions. In the context of POD-Galerkin modeling based on adaptive finite element snapshots, the main challenge is to find suitable continuous extensions of the Dirichlet data, which need to be subtracted from the snapshots before a POD basis is computed. For the derivation of a velocity POD-Galerkin model, we must ensure that these continuous extensions fulfill the correct weak divergence-free property.

Inhomogeneous problem setting

We extend (4.44) to the case of inhomogeneous Dirichlet boundary conditions by introducing Dirichlet boundary data $y_D : [0,T] \times \partial\Omega \to \mathbb{R}$. The resulting problem reads as follows: find a velocity field $\breve{y}$ and a pressure field p such that

$$\begin{aligned}
\breve{y}_t + (\breve{y}\cdot\nabla)\breve{y} - Re^{-1}\Delta\breve{y} + \nabla p &= f && \text{in } (0,T]\times\Omega, && (4.57a)\\
\nabla\cdot\breve{y} &= 0 && \text{in } [0,T]\times\Omega, && (4.57b)\\
\breve{y} &= y_D && \text{on } [0,T]\times\partial\Omega, && (4.57c)\\
\breve{y}(0,\cdot) &= y_0 && \text{in } \Omega. && (4.57d)
\end{aligned}$$

where $\nabla \cdot y_0 = 0$ in Ω and $y_0 = y_D(0,\cdot)$ on $\partial\Omega$. In the following, we derive a homogenized version of this problem, which provides a foundation for the subsequent finite element discretization and reduced-order modeling.

Homogenized problem setting

We assume that the boundary function y_D is sufficiently smooth such that it can be continuously extended by a function $g : [0,T] \times \bar{\Omega} \to \mathbb{R}$ with $g(t,\cdot)|_{\partial\Omega} = y_D(t,\cdot)$ for $t \in [0,T]$. The regularity requirements on g are such that a unique weak solution exists, see [154]. We provide a concrete choice of g by computation.

We homogenize (4.57) by subtracting the boundary function from the inhomogeneous velocity solution such that the homogeneous velocity field is given by $y = \breve{y} - g$. Substituting $\breve{y}$ in (4.57), we obtain the following homogenized problem: find a velocity y and a pressure p such that

$$\begin{aligned}
y_t + (y\cdot\nabla)y + (g\cdot\nabla)y + (y\cdot\nabla)g - \mathit{Re}^{-1}\Delta y + \nabla p & & \quad (4.58\text{a})\\
= f - (g\cdot\nabla)g + \mathit{Re}^{-1}\Delta g - g_t & \quad \text{in } (0,T]\times\Omega, & \quad (4.58\text{b})\\
\nabla\cdot y = -\nabla\cdot g & \quad \text{in } [0,T]\times\Omega, & \quad (4.58\text{c})\\
y = 0 & \quad \text{on } [0,T]\times\partial\Omega, & \quad (4.58\text{d})\\
y(0,\cdot) = y_0 - g & \quad \text{in } \Omega, & \quad (4.58\text{e})
\end{aligned}$$

where $\nabla \cdot y_0 = 0$ in Ω and $y_0 = g(0,\cdot)$ on $\partial\Omega$.

In order to derive a time-discrete weak form of the homogenized problem, we implement the time integrals involving the Dirichlet data using the right-sided rectangle rule, which evaluates the Dirichlet data at the new time instance. For ease of notation, we define $f^j := f(t_j)$ and $g^j := g(t_j)$ for $j = 0,\dots,n$. Consequently, the time-discrete weak form of the homogenized problem consists in finding sequences $y^1,\dots,y^n \in H_0^1(\Omega)$ and $p^1,\dots,p^n \in L_0^2(\Omega)$, for given $y^0 = y_0 - g^0$ with $y_0 \in H_{div}(\Omega)$, such that

$$\left(\frac{y^j - y^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} + c(y^j,y^j,v) + c(g^j,y^j,v) + c(y^j,g^j,v) + a(y^j,v) + b(v,p^j) \quad (4.59\text{a})$$

$$= \langle f^j, v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} - c(g^j,g^j,v) - a(g^j,v) - \left(\frac{g^j - g^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} \quad \forall v \in H_0^1(\Omega), \quad (4.59\text{b})$$

$$b(y^j,q) = -b(g^j,q) \quad \forall q \in L_0^2(\Omega). \quad (4.59\text{c})$$

We apply an adaptive finite element method. Let us denote by V_h^0 the velocity finite element space associated with the initial mesh $\mathcal{T}_h^0$. For a given initial condition $(y_h^0, v)_{L^2(\Omega)} = (y_0 - g^0, v)_{L^2(\Omega)}$ for all $v \in V_h^0$ with $y_0 \in H_{div}(\Omega)$ the fully discrete homogenized Navier–Stokes problem reads as follows: find $y_h^1 \in V_h^1,\dots,y_h^n \in V_h^n$ and $p_h^1 \in Q_h^1,\dots,p_h^n \in Q_h^n$ such that

$$\left(\frac{y_h^j - y_h^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} + c(y_h^j,y_h^j,v) + c(g^j,y_h^j,v) + c(y_h^j,g^j,v) + a(y_h^j,v) + b(v,p_h^j) \quad (4.60\text{a})$$

$$= \langle f^j, v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} - c(g^j,g^j,v) - a(g^j,v) - \left(\frac{g^j - g^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} \quad \forall v \in V_h^j, \quad (4.60\text{b})$$

$$b(y_h^j,q) = -b(g^j,q) \quad \forall q \in Q_h^j. \quad (4.60\text{c})$$

Lifting function

Based on (4.60), approximations to the inhomogeneous solutions $\breve{y}(t_j)$ of (4.57) are given by $\hat{y}_h^j := y_h^j + g^j$ for $j = 0, \dots, n$. Regardless of the choice of lifting functions $g^0, \dots, g^n$, we can guarantee that $\hat{y}_h^0$ fulfills the initial condition (4.57d) and $\hat{y}_h^1, \dots, \hat{y}_h^n$ fulfill the Dirichlet condition (4.57c) by construction. Nevertheless, in order to solve (4.60) numerically, candidates of $g^0, \dots, g^n$ must be fixed, at least implicitly. Our approach to reduced-order modeling is not tied to a particular choice. In the following, we provide suitable candidates which can be realized without the need to modify usual finite element codes.
Note that for the velocity finite element spaces we have $V_h^j \subset H_0^1(\Omega)$. Thus, for the context of inhomogeneous Dirichlet conditions, we start by introducing the spaces V_D^j for $j = 1, \dots, n$, which denote the spaces spanned by the union of the finite element basis functions of V_h^j and the finite element basis functions associated with the corresponding Dirichlet boundary nodes. We assume that in (4.60) the integrals involving g^j are approximated by a numerical quadrature which consists of substituting the Lagrange interpolation of g^j onto V_D^j and integrating the resulting piecewise polynomials exactly. We assume that by a finite number of refinements of any $V_D^1, \dots, V_D^n$ one can find a reference finite element space $\tilde{V}_D$ such that $V_D^1, \dots, V_D^n \subset \tilde{V}_D$. Now, for $j = 1, \dots, n$, we define lifting functions g^j as a sufficiently smooth continuous extension of the Dirichlet data $y_D(t_j)$ into the domain Ω such that g^j is zero at all nodes of the reference finite element space $\tilde{V}$. This is equivalent to the standard approach of using an *approximate* Dirichlet lifting given by a Lagrange interpolation of the Dirichlet data onto the finite element space at the boundary and a subsequent continuous extension using the finite element space in the interior, because we have

$$\begin{cases} g^j = y_D(t) & \text{at all Dirichlet nodes of } V_D^j, \\ g^j = 0 & \text{at all interior nodes of } V_D^j. \end{cases}$$

A disadvantage of the standard approach is that it implies a Dirichlet lifting which satisfies the boundary data only in an approximate sense. Our description, on the other hand, delivers an output which is exact at the boundary. In particular, we have

$$\begin{cases} \hat{y}_h^j = y_h^j & \text{at all interior nodes of } \tilde{V}_D, \\ \hat{y}_h^j = g^j & \text{at all Dirichlet nodes of } \tilde{V}_D, \\ y_h^j = 0 & \text{at all Dirichlet nodes of } \tilde{V}_D. \end{cases}$$

This holds for all $\tilde{V}_D$ which fulfill our assumptions, without the need to specify a concrete candidate of $\tilde{V}_D$ during the adaptive finite element simulation. When the adaptive finite element simulation is finished and $V_D^1, \dots, V_D^n$ are available, some $\tilde{V}_D$ can be computed by refinement and g^j can be evaluated at all nodal points of $\tilde{V}_D$. Therefore, we are even able to formulate a finite element discretization of (4.59) on $(\tilde{V}, \tilde{Q})$ using the same $g^0, \dots, g^n$ as in (4.60). Moreover, we are able to solve (4.59) on subspaces of $(\tilde{V}, \tilde{Q})$ using the same $g^0, \dots, g^n$ as in (4.60).

Remark 4.7. *In principle, it is possible to impose a weak divergence-free constraint on the homogenized velocity finite element solution by using lifting functions which are computed such that $b(g^j, q) = 0$ for all $q \in Q_h^j$ for $j = 0, \dots, n$. But this would require the solution of an additional stationary finite element problem for each g^j. Moreover, this would not automatically imply a weak divergence-free property with respect to a reference pressure space $\tilde{Q}$. An alternative to the implicit choice of the Dirichlet lifting function is its explicit choice at the level of the strong formulation* (4.58)*. Disadvantages would be a possibly larger support of such a lifting function and the effort of actually finding a suitable function. Also in this case, it would be attractive to impose a strong divergence-free constraint on g^j, because this implies a weakly divergence-free homogenized velocity field y_h^j. Still, finding a suitable candidate may be challenging in general.*

Velocity reduced-order model for the inhomogeneous setting

In the following, we derive a reduced-order model for the velocity field, based on the semi-discretized problem (4.59). We introduce the projections $P_g^{\mathcal{V},\mathcal{Q}}$ according to Problem 4.5 by

Problem 4.8. *For given $u \in X$, sufficiently smooth g and given spaces $\mathcal{V}$ and $\mathcal{Q}$, find $P_g^{\mathcal{V},\mathcal{Q}}(u)$ which solves*

$$\min_{v\in\mathcal{V}} \frac{1}{2}\|v-u\|_X^2 \quad \textit{subject to} \quad b(v+g,q)=0 \quad \forall q\in\mathcal{Q}.$$

We use $X=\mathcal{V}=H_0^1(\Omega)$ and $\mathcal{Q}=L_0^2(\Omega)$ in the following. By definition of this projection onto a divergence-free space, we have $b(P_{g^j}^{\mathcal{V},\mathcal{Q}}(0)+g^j,q)=0$ for all $q\in L_0^2(\Omega)$ and $j=0,\dots,n$. In (4.59), we substitute $g^j=g^j+P_{g^j}^{\mathcal{V},\mathcal{Q}}(0)-P_{g^j}^{\mathcal{V},\mathcal{Q}}(0)$ and reformulate the equations so that we can set $\tilde{y}^j=y^j-P_{g^j}^{\mathcal{V},\mathcal{Q}}(0)$. We obtain the following problem, which is equivalent to (4.59): for given $\tilde{y}^0=y_0-\tilde{g}^0$ with $y_0\in H_{div}(\Omega)$ and $\tilde{g}^j=g^j+P_{g^j}^{\mathcal{V},\mathcal{Q}}(0)$ for $j=0,\dots,n$, find $\tilde{y}^1,\dots,\tilde{y}^n\in H_0^1(\Omega)$ and $p^1,\dots,p^n\in L_0^2(\Omega)$ such that

$$\left(\frac{\tilde{y}^j-\tilde{y}^{j-1}}{\Delta t_j},v\right)_{L^2(\Omega)}+c(\tilde{y}^j,\tilde{y}^j,v)+c(\tilde{g}^j,\tilde{y}^j,v)+c(\tilde{y}^j,\tilde{g}^j,v)+a(\tilde{y}^j,v)+b(v,p^j) \tag{4.61a}$$

$$=\ \langle f^j,v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)}-c(\tilde{g}^j,\tilde{g}^j,v)-a(\tilde{g}^j,v)-\left(\frac{\tilde{g}^j-\tilde{g}^{j-1}}{\Delta t_j},v\right)_{L^2(\Omega)} \qquad \forall v\in H_0^1(\Omega), \tag{4.61b}$$

$$b(\tilde{y}^j,q)=\ 0 \qquad \forall q\in L_0^2(\Omega). \tag{4.61c}$$

One can show that (4.60) is a discretization of (4.61) by replacing $(H_0^1(\Omega),L_0^2(\Omega))$ with (V_h^j,Q_h^j) for $j=1,\dots,n$ in (4.61). Then, for the resulting solution holds $\tilde{y}^j=y_h^j-P_{g^j}^{V_h^j,Q_h^j}$ if $\tilde{g}^j=g^j+P_{g^j}^{V_h^j,Q_h^j}(0)$ for $j=0,\dots,n$. In this way, (4.60) can be used to obtain approximate solutions of (4.61).

We base the model equation on a discretization of (4.61) using the pair of reference spaces $(\tilde{V},\tilde{Q})$ as test spaces together with $\tilde{g}^j=g^j+P_{g^j}^{\tilde{V},\tilde{Q}}(0)$. The resulting solutions are approximations to the solutions of (4.61) using the original pair of spaces $(H_0^1(\Omega),L_0^2(\Omega))$. We have shown that the solutions of (4.61) using the original pair of spaces $(H_0^1(\Omega),L_0^2(\Omega))$ are approximated by $y_h^j-P_{g^j}^{V_h^j,Q_h^j}(0)$ for $j=1,\dots,n$ resulting from (4.60). But these solutions are not weakly divergence-free with respect to the reference pair of spaces $(\tilde{V},\tilde{Q})$. Therefore, we have to modify them.

Following the velocity-ROM approach a) in Section 4.6.3 we substitute $y_h^j-P_{g^j}^{V_h^j,Q_h^j}(0)$ by their approximations $P_{g^j}^{\tilde{V},\tilde{Q}}(y_h^j)-P_{g^j}^{\tilde{V},\tilde{Q}}(0)$ for $j=1,\dots,n$. Using now $P_{g^j}^{\tilde{V},\tilde{Q}}(y_h^j)-P_{g^j}^{\tilde{V},\tilde{Q}}$ as snapshots in a POD yields POD basis functions

$$\psi_i\in\operatorname{span}(P_{g^1}(y_h^1)-P_{g^1}(0),\dots,P_{g^n}(y_h^n)-P_{g^n}(0))\subset\tilde{V}_{div} \qquad \forall i=1,\dots,\ell_y$$

for some $\ell_y\le n$, which define a POD space $V_\ell:=\operatorname{span}(\psi_1,\dots,\psi_{\ell_y})\subset\tilde{V}_{div}$.

In the time-discrete equation (4.61), we use the pair $(V_\ell,\tilde{Q})$ as test and trial spaces. Consequently, the continuity equation is fulfilled by construction. For the pressure term, we have $b(v,p^j)=0$ for all $v\in V_\ell$ and all $p^j\in\tilde{Q}$. The resulting reduced-order model is given by the following set of equations: for $y_\ell^0 = y_0 - \tilde{g}^0$ with $y_0 \in H_{div}(\Omega)$ and $\tilde{g}^j = g^j + P_{g^j}^{\tilde{V},\tilde{Q}}(0)$ for $j = 0, \dots, n$, find $y_\ell^1, \dots, y_\ell^n \in V_\ell$ such that

$$\begin{aligned}\left(\frac{y_\ell^j - y_\ell^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} &+ c(y_\ell^j, y_\ell^j, v) + c(\tilde{g}^j, y_\ell^j, v) + c(y_\ell^j, \tilde{g}^j, v) + a(y_\ell^j, v) \\ &= \langle f^j, v\rangle_{H^{-1}(\Omega), H_0^1(\Omega)} - c(\tilde{g}^j, \tilde{g}^j, v) - a(\tilde{g}^j, v) - \left(\frac{\tilde{g}^j - \tilde{g}^{j-1}}{\Delta t_j}, v\right)_{L^2(\Omega)} \quad \forall v \in V_\ell.\end{aligned} \tag{4.62}$$

Remark 4.9. *Concerning the computational complexity, we have to additionally consider the projections $P_{g^j}^{\tilde{V},\tilde{Q}}(0)$ for $j = 0, \dots, n$ in comparison to the homogeneous case. Therefore, the solution of Problem 4.5 has to be computed $n+1$ times additionally to the projections of the homogeneous solutions y_h^j.*

Following the velocity-ROM approach b) in Section 4.6.3, we first introduce a set of modified homogeneous solutions $\hat{y}^j - P_{g^j}^{\tilde{V},\tilde{Q}}(0)$, for $j = 1, \dots, n$. These modified snapshots can be constructed using e.g. a Lagrange interpolation of the original homogeneous solutions y_h^j onto the reference space $\tilde{V}$ and an approximation of $P_{g^j}^{Vj,Q^j}(0)$ for $j = 1, \dots, n$. From these modified snapshots, we compute a POD basis $\hat{\psi}_1, \dots, \hat{\psi}_{\ell_y} \in \tilde{V}$. Note that these modes are in general not discrete divergence-free. Thus, they are then projected onto the space $\tilde{V}_{div}$ by solving Problem 4.6 with $g = 0$. This leads to a divergence-free velocity POD space $V_\ell = \operatorname{span}(\psi_1, \dots, \psi_{\ell_y}) \subset \tilde{V}_{div}$. Replacing $(H_0^1(\Omega), L_0^2(\Omega))$ by the pair $(V_\ell, \tilde{Q})$ in (4.61) leads to a reduced-order model of the form (4.62).

Velocity-pressure reduced-order model for the inhomogeneous setting

To derive a velocity-pressure reduced-order model of the homogenized problem (4.60), we require a suitable inf-sup stable pair of reduced spaces. Since the homogenization does not alter the bilinear form $b(\cdot,\cdot)$, the inf-sup stability criterion stays the same. Therefore, we compute a pressure reduced space Q_ℓ and a velocity reduced space V_ℓ like in Section 4.6.4, but using Lagrange interpolated velocity and pressure snapshots of (4.60) instead of (4.48). We derive a stable POD-Galerkin model from the time-discrete problem (4.59) by using the pair (V_ℓ, Q_ℓ) as test and trial spaces.

We solve the reduced-order model for the POD approximations $y_\ell^1, \dots, y_\ell^n$ of the homogeneous velocity fields and the POD approximations $p_\ell^1, \dots, p_\ell^n$ of the pressure fields. Finally, $y_\ell^j + g^j$ is a time-discrete reduced-order approximation of the velocity solution of the inhomogeneous problem.

4.6.6 Numerical example

We consider the benchmark problem of a lid-driven cavity flow as a numerical example. The implementation is done in `C++` and we use FEniCS [13, 132] for the solution of the differential equations using finite elements and ALBERTA [161] for dealing with hierarchical meshes. All simulations are performed on a compute server with 512 GB RAM.

Problem setting

As spatial domain, we consider the open bounded unit square $\Omega = (0,1) \times (0,1) \subset \mathbb{R}^2$ and set the end time $T = 1$. The governing equations are the Navier–Stokes equations with inhomogeneous Dirichlet conditions given in (4.57). We utilize a regularized lid-velocity according to [76, 112], which implies $\nabla \cdot y = 0$ in the corners. An additional regularization in time allows for a smooth start up from $y_0(x) = 0$. In particular, the Dirichlet data is given as $y_D(t,x) = y_D^t(t) y_D^x(x)$ for all $(t,x) \in [0,1] \times \partial\Omega$, where

$$y_D^t(t) = \begin{cases} 1 - \frac{1}{4}(1-\cos((0.1-t)\pi/0.1))^2 & \text{if } t \in [0, 0.1), \\ 1 & \text{if } t \in [0.1, 1], \end{cases}$$

$$y_D^x(x) = \begin{cases} 1 - \frac{1}{4}(1-\cos((0.1-x_1)\pi/0.1))^2 & \text{if } x_2 = 1, x_1 \in [0, 0.1], \\ 1 & \text{if } x_2 = 1, x_1 \in (0.1, 0.9), \\ 1 - \frac{1}{4}(1-\cos((x_1-0.9)\pi/0.1))^2 & \text{if } x_2 = 1, x_1 \in [0.9, 1], \\ 0 & \text{otherwise.} \end{cases}$$

As body force we choose $f(t,x) = 0$ and consider laminar flow with a Reynolds number $Re = 100$.

Finite element results

The initial triangulation $\mathcal{T}_h^{init}$ is chosen as a criss-cross triangulation with 9 node points in both x_1- and x_2-directions, see Figure 48 (left). For the time discretization, we utilize a uniform time grid with time step size $\Delta t = 0.01$ leading to $n = 100$ time points. In the h-adaptive finite element Algorithm 1 we choose as stopping tolerance $\varepsilon = 0.01$ and as Dörfler marking parameter $\theta = 0.1$.

The resulting adaptive finite element velocity components and pressure solutions as well as the adapted meshes are shown in Figure 49 for three different time instances. The maximal number of node points in the adapted meshes is 3287. The overlay mesh (shown in Figure 48, right) has 3729 node points. For comparison, an uniform discretization with fineness as small as the smallest triangle in the adapted meshes would have 131585 node points, which leads to an *out of memory* error with our computing machines.

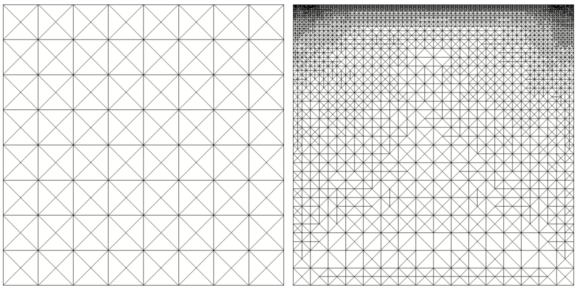

Figure 48: Initial triangulation $\mathcal{T}_h^{init}$ (left) and overlay of all adapted meshes (right)

POD reduced-order solutions

In order to compute a POD basis, we choose in (2.7) for the velocity POD the space $X = H_0^1(\Omega)$ and for the pressure POD the space $X = L^2(\Omega)$. As time weights, we choose $\alpha_j = \Delta t$, i.e. we interpret the sum in (2.7) as a quadrature of the time integral in (2.5) with a right-sided rectangle quadrature rule. This complies with the interpretation of the implicit Euler scheme as a discontinuous Galerkin method.
For the methodologies described in Sections 4.6.3 and 4.6.4, we choose the reference velocity and pressure spaces $(\tilde{V}, \tilde{Q})$ as the Taylor–Hood finite element pair associated with the overlay of the adapted finite element meshes, which is shown in Figure 48 (right). In Problem 4.5, we choose $X = H_0^1(\Omega)$ in order to enable a consistent setting.
The first three velocity POD modes (without divergence-free projection) as well as the pressure POD modes and associated supremizer functions are displayed in Figure 50. Note that the POD modes computed from weakly divergence-free projected snapshots as well as the projected POD modes look very similar to the POD modes in Figure 50.

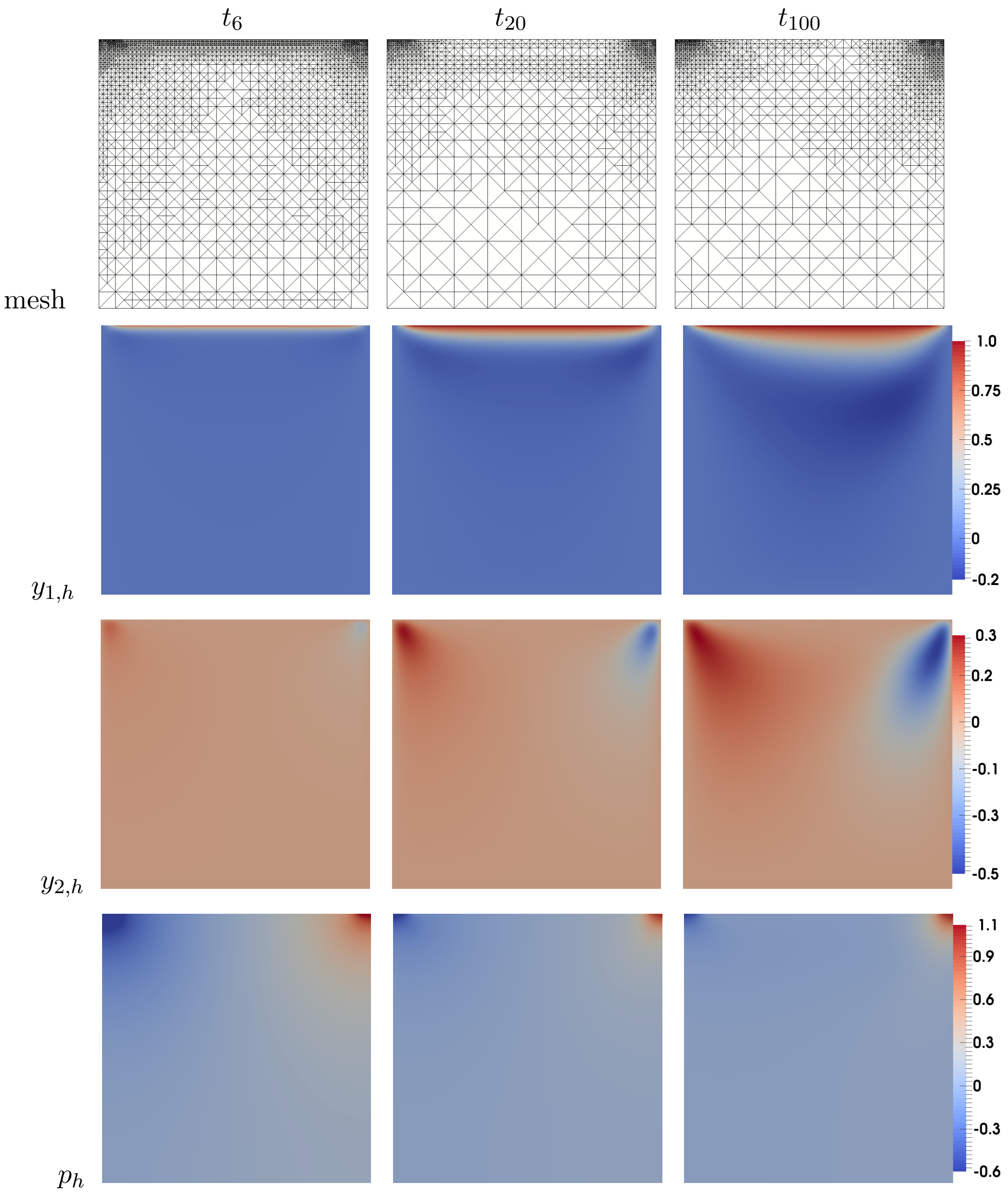

Figure 49: Adapted finite element meshes, velocity components $y_h = (y_{1,h}, y_{2,h})$ and pressure solutions at time instances t_n for $n = 6, 20, 100$

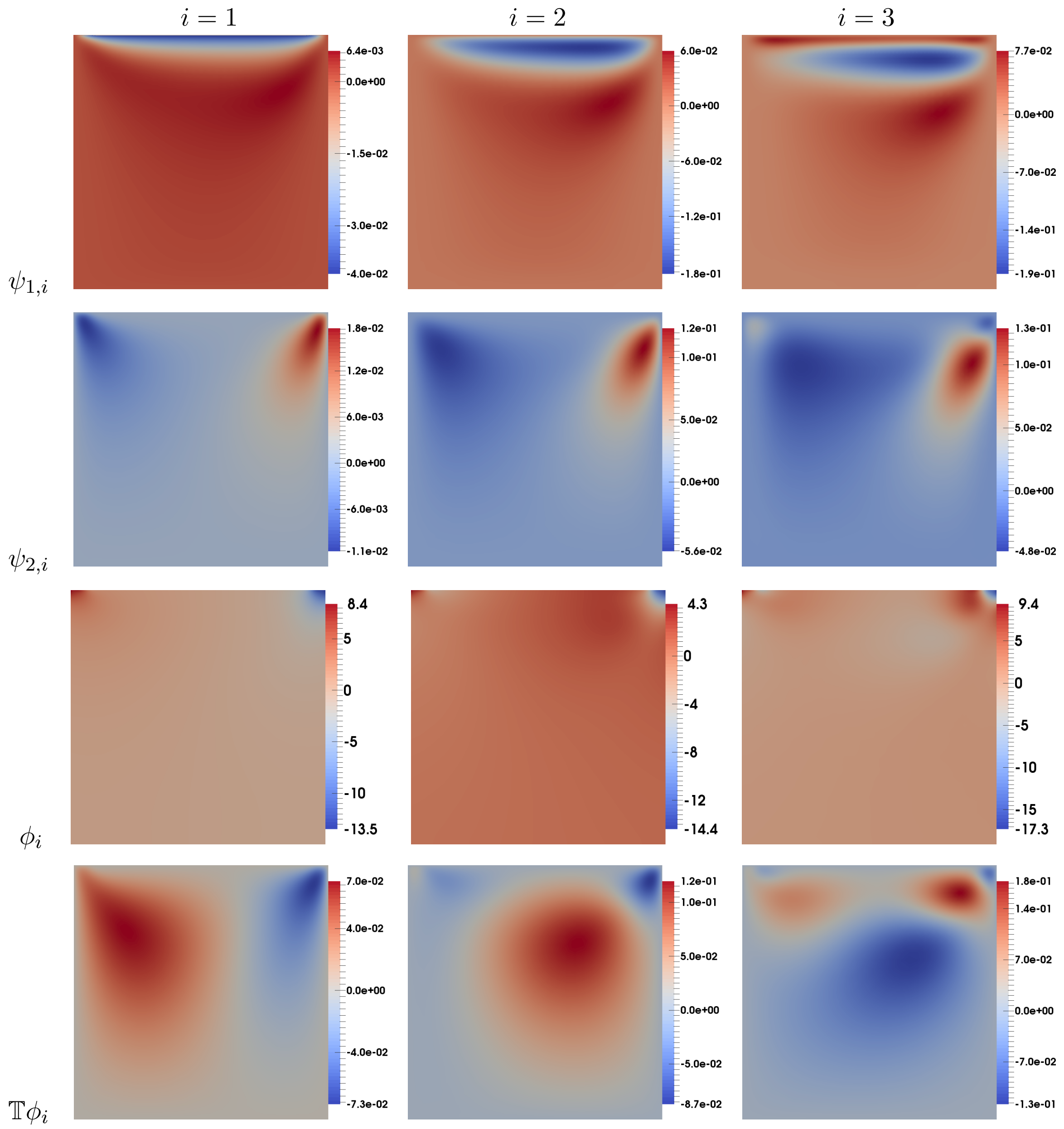

Figure 50: First three POD modes $i = 1, 2, 3$ for the velocity $\psi_i = (\psi_{1,i}, \psi_{2,i})$ and the pressure ϕ_i, first three supremizer functions $\mathbb{T}\phi_i$

We compare the approaches of a velocity reduced-order modeling and the velocity-pressure reduced-order modeling based on supremizer stabilization concerning accuracy and efficiency. In the velocity-pressure approach we choose for simplicity the same number of velocity POD basis modes as pressure POD basis functions. A short-hand notation for the considered reduced-order approximations is given in Table 7.

In order to validate the approximation quality of the POD reduced-order solution we consider the following relative error for the velocity

$$err_{FE-POD} = \frac{\|y_h - y_\ell\|_{L^2(0,T;\Omega)}}{\|y_h\|_{L^2(0,T;\Omega)}}.$$

The results are shown in Figure 51. First of all, we observe that a velocity-pressure reduced-order model without supremizer stabilization leads to an unstable model and unsatisfactory approximation results. A naive approach, i.e. a velocity reduced-order model without utilizing a divergence-free projection leads to better approximation results in this example. From $\ell_y = 1$

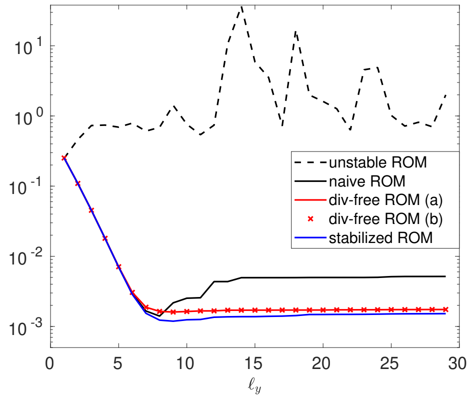

Figure 51: Relative velocity errors of different reduced-order approximations according to Table 7 and different numbers of utilized velocity POD basis functions

method	description
unstable ROM	velocity-pressure reduced-order (4.50), but no supremizers
naive ROM	velocity reduced-order model (4.51), but using Lagrange interpolations instead of divergence-free projections of the FE solutions
div-free ROM (a)	velocity reduced-order model according to approach a) in Section 4.6.3
div-free ROM (b)	velocity reduced-order model according to approach b) in Section 4.6.3
stabilized ROM	velocity-pressure reduced-order model according to Section 4.6.4

Table 7: Short-hand notation for the different reduced-order models

to $\ell_y = 8$, the relative error between the finite element solution and the naive reduced-order approximation decreases. However, for $\ell_y > 8$, the error starts to increase, which is not what we would like to achieve.

Now, let us compare the velocity reduced-order approaches of Section 4.6.3 with the velocity-pressure reduced-order approach of Section 4.6.4. All of these methodologies lead to stable results in the sense that the error between the POD reduced-order model and the finite element solution decreases for an increasing number of utilized POD modes up to $\ell_y = 8$. For $\ell_y > 8$, the error stagnates. Both velocity approaches (a) and (b) lead to very similar results. In comparison to the velocity model, the accuracy of the velocity-pressure model is slightly better.

	div-free ROM (b)	stabilized ROM
FE solution	125.66	125.66
offline times		
→ construction of overlay mesh	3.81	3.81
→ velocity POD	1.73	1.73
→ pressure POD	–	0.05
→ div-free projection of POD modes	5.54	–
→ supremizers	–	0.31
ROM solution	0.009	0.03

Table 8: Computation times (sec) for finite element solution, offline and online times using $\ell_y = 8$ POD modes

Concerning the computational efficiency, we compare the calculation times of the velocity reduced-order approach (b) with the velocity-pressure reduced-order approach. The velocity reduced-order approach (a) is more expensive than the approach (b), since Problem 4.5 has to be solved for each snapshot, i.e. n times, whereas in the latter approach it only has to be solved for each velocity POD mode, i.e. $\ell_y < n$ times. For comparison, we list the computational time for the adaptive finite element solution in Table 8. It turns out that the divergence-free projection of the POD modes is more expensive than the computation of the supremizer functions. Concerning the reduced-order online time, the solution of the velocity model is very fast with only 0.009 seconds which leads to a speed up of factor 13962 compared to the adaptive full-order finite element simulation. The solution of the velocity-pressure reduced order model takes 0.03 seconds, so it is slower than the velocity reduced-order model. This is due to the fact that the dimension of the velocity model depends solely on ℓ_y, whereas the complexity of the velocity-pressure reduced-order model additionally depends on the number of supremizer functions and the number of pressure POD basis functions. Recall that in this test run, we take for simplicity the same number of POD modes for the pressure as for the velocity, i.e. $\ell_p = \ell_y$.

These factors motivate to utilize POD reduced-order modeling in a multi-query setting. We note that a finite element simulation on a grid with the same fineness as the smallest triangle as in the adapted meshes is not possible on our computing machines since it reaches the memory limit.

4.6.7 Further directions of research

In order to avoid the difficulty that the divergence constraint of the adaptive finite element solution only holds true in the current finite element discretization, an option could be to utilize finite element methods which guarantee by construction that the discrete Navier-Stokes solutions are divergence-free in the strong sense. In [113, 130], different approaches in order to tackle the lack of pressure-robustness which stems from enforcing the divergence constraint only discretely are compared. In particular, inf-sup stable conforming element pairs $V_h \subset H_0^1(\Omega)$, $Q_h \subset L_0^2(\Omega)$ are derived which satisfy $\nabla \cdot V_h \subset Q_h$. As a consequence, it can be proved that $V_{h,div} \subset H_{div}$. The derivation of a POD reduced-order model for this setting as well as a comparison with respect to efficiency and accuracy of the proposed methods of this section could be carried out in future research.

Moreover, we aim to utilize the proposed methodology for POD reduced-order modeling of incompressible fluid flows in a multi-query setting like uncertainty quantification or optimal control. There, we expect large speed up factors in the offline phase in comparison to the use of a static uniform discretization as well as a speed up in the online phase whenever new full-order solutions are required in order to update the reduced-order basis.

4.7 POD model order reduction with space-adapted snapshots in optimal control

In this section, we apply the techniques from the previous Sections 4.2 to 4.5 for the solution of a multi-query scenario. We consider the optimal control of a convective Cahn–Hilliard system, where the control enters through the velocity in the transport term. We prove the existence of a solution to the considered optimal control problem. For the numerical solution, we replace the full-order model by POD reduced-order surrogates which are computed from space-adapted snapshots.

Literature overview

Let us give an overview of literature concerning optimal control of phase field models and the reduced-order modeling thereof.

In [52] an optimal control problem of a coupled temperature phase field model with polynomial free energy is considered. Existence of a solution to the considered optimal control problem is proved and first-order optimality conditions are derived. In a numerical example, the shape of a phase field is controlled. In [93] the setting of [52] is extended by introducing control constraints. An efficient solver for the linearized state and adjoint equations is constructed which is based on a projected Newton method. In [185] a POD reduced-order model for the problem setting of [93] is considered. For the numerical solution of the optimal control system an augmented Lagrangian–SQP method is used and the high-fidelity solutions on static meshes are replaced by POD reduced-order approximations.

In [92] a phase field approach is used in order to control the shape evolution of single layer islands on a crystalline surface. The control goal is to find an optimal electric field which steers the initial phase shape as close as possible to a given desired shape while keeping the magnitude of the electric field low. The control enters the Cahn–Hilliard model through (2.18b). A proof of concept is given in a numerical test run which moves and transforms an initial shape in horizontal direction.

An analysis of the optimal control of the Cahn–Hilliard model with distributed control is given in [97]. Existence of a solution in case of a Moreau–Yosida relaxed double-obstacle free energy is proved and a convergence analysis of optimal controls of the regularized problem to an optimal control of the original problem with a double-obstacle free energy is carried out. Moreover, C-stationarity conditions are derived.

In [158] the optimal control of a non-local Cahn–Hilliard model with degenerate mobility and singular potential is considered in 3d. The control enters the system through the velocity in the transport term. Existence of an optimal control is proved and first-order optimality conditions are derived.

The optimal control problem of [97] with distributed control which enters the Cahn–Hilliard system on the right hand side of (2.18a) is considered in [5] within a trust-region POD framework. The numerical experiments are based on [85] and use spatially adapted finite element snapshots. The optimization goal consists of deforming a circular shape into a rounded square. Different choices for the snapshot set are considered such as a separate POD basis for the state and adjoint states as well as a combined snapshot set of the system variables. Similar results are published in a collaboration with Jan Oke Alff and Michael Hinze in [6].

Note that some of the results of this section are based on a collaboration with Michael Hinze and Nicolas Scharmacher in [87].

4.7.1 Optimal control of a convective Cahn–Hilliard system

Let $\Omega \subset \mathbb{R}^d, d \in \{2,3\}$ denote an open bounded domain with Lipschitz boundary $\partial\Omega$, see Definition 2.17. Let $T > 0$ be a fixed end time. We consider the following optimal control problem of a convective Cahn–Hilliard system: minimize the quadratic cost functional given by

$$J(\varphi, u) = \frac{\beta_1}{2}\|\varphi - \varphi_d\|^2_{L^2(0,T;\Omega)} + \frac{\beta_2}{2}\|\varphi - \varphi_T\|^2_{L^2(\Omega)} + \frac{\gamma}{2}\|u\|^2_U, \tag{4.63}$$

subject to the Cahn–Hilliard system with transport

$$\varphi_t + (\mathcal{B}u)\cdot\nabla\varphi - b\Delta\mu = 0 \quad \text{in } (0,T]\times\Omega, \tag{4.64a}$$

$$-\sigma\varepsilon\Delta\varphi + \frac{\sigma}{\varepsilon}\mathcal{F}'(\varphi) = \mu \quad \text{in } (0,T]\times\Omega, \tag{4.64b}$$

and

$$u \in U_{ad}. \tag{4.65}$$

We denote by φ the phase field, μ is the chemical potential, $\mathcal{F}$ is the free energy function, b is a constant mobility, σ is the surface tension and ε denotes the interface parameter. For more details on the Cahn–Hilliard system we refer to Section 2.5. In (4.63) we denote by $\beta_1, \beta_2 \geq 0$ given constants, $\gamma > 0$ is the penalty parameter, $\varphi_d \in L^2(0,T;\Omega)$ is the desired phase field and $\varphi_T \in L^2(\Omega)$ is the target phase pattern at final time.

Motivated by applications where it is only possible to control in certain prescribed areas within the spatial domain, we consider located control which is a time-dependent quantity and matched to the time-space domain via a control operator $\mathcal{B}$. In particular, we consider the control space $U = L^2(0,T;\mathbb{R}^m)$, with $m \in \mathbb{N}$ and introduce an admissible set of controls by

$$u \in U_{ad} = \{u \in U \mid u_a(t) \leq u(t) \leq u_b(t) \text{ in } \mathbb{R}^m \text{ a.e. in } [0,T]\}$$

with $u_a, u_b \in L^\infty(0,T;\mathbb{R}^m), u_a(t) \leq u_b(t)$ almost everywhere in $[0,T]$. The inequalities between vectors are understood componentwise. We define the control operator as

$$\mathcal{B} : U \to L^\infty(0,T;L^\infty(\Omega)) \cap L^2(0,T;H^1(\Omega)), \quad u \mapsto \mathcal{B}u = \sum_{i=1}^m u_i \chi_i,$$

where $\chi_i \in L^\infty(\Omega) \cap H^1(\Omega)$ for $i = 1, \ldots, m$ denote given shape functions. Note that we use the same notation for scalar- and vector-valued functions, meaning that all components of a vector-valued function belong to the corresponding scalar function space.

We equip (4.64) with homogeneous Neumann boundary conditions

$$\frac{\partial \varphi}{\partial \vec{n}} = 0 \quad \text{and} \quad \frac{\partial \mu}{\partial \vec{n}} = 0 \quad \text{on } [0,T] \times \partial\Omega, \tag{4.66}$$

and impose an initial condition for the phase field

$$\varphi(0,\cdot) = \varphi_0 \quad \text{in } \Omega. \tag{4.67}$$

The aim of the optimal control problem is to steer a given initial phase distribution φ_0 to a given desired phase pattern φ_d during the time $(0,T)$ and φ_T at end time T. This problem can also be interpreted as an optimal control of a free boundary which is encoded through the phase field variable.
We assume that the initial phase field φ_0 is in $H^1_{(0)}(\Omega) = H^1(\Omega) \cap L^2_{(0)}(\Omega)$ with $L^2_{(0)}(\Omega) = \{f \in L^2(\Omega) : \frac{1}{|\Omega|} \int_\Omega f dx = g\}$ with mean value $g = 0$ (w.l.o.g.). Moreover, we assume that $\mathcal{E}_0 = \mathcal{E}(\varphi_0) < \infty$ where $\mathcal{E}$ denotes the Ginzburg–Landau free energy defined in (2.15).

Note that in (4.64) the convection models a possible flow of the phase field system at a given velocity field $\mathcal{B}u$, for which we impose a no-flow condition on the boundary in order to preserve mass conservation. In particular, we assume $\chi_i \in H_{div}(\Omega)$ for $i = 1, \ldots, m$ with H_{div} introduced in (4.45) as the space of solenoidal vector fields by

$$H_{div}(\Omega) := \{v \in L^2(\Omega) : \nabla \cdot v = 0, v \cdot \vec{n}|_{\partial\Omega} = 0 \text{ in the sense of traces}\}.$$

Moreover, we will need the space

$$H^1_n(\Omega) = \{f \in H^1(\Omega) : f \cdot \vec{n}|_{\partial\Omega} = 0 \text{ in the sense of traces}\}.$$

Note that in the context of multiphase flow, the transport term represents the coupling to the Navier–Stokes system, see e.g. [2, 104].

The weak form of the state equation reads as follows: for a given initial phase field φ_0 find a phase field φ with $\varphi(0,\cdot) = \varphi_0$ and a chemical potential μ such that

$$\langle \varphi_t(t), v\rangle_{H^{-1}(\Omega),H^1_0(\Omega)} + ((\mathcal{B}u(t)) \cdot \nabla\varphi(t), v)_{L^2(\Omega)} + b(\nabla\mu(t), \nabla v)_{L^2(\Omega)} = 0 \qquad \forall v \in H^1(\Omega), \tag{4.68a}$$

$$\sigma\varepsilon(\nabla\varphi(t), \nabla v)_{L^2(\Omega)} + \frac{\sigma}{\varepsilon}(\mathcal{F}'(\varphi(t)), v)_{L^2(\Omega)} = (\mu(t), v)_{L^2(\Omega)} \quad \forall v \in H^1(\Omega), \tag{4.68b}$$

holds for almost all $t \in (0, T)$.

It is shown in [1] for $\sigma = b = \varepsilon = 1$ that the state equation (4.64) is well-posed. This result is obtained for a free energy with smooth convex part $\mathcal{F}_+ \in C([-1, 1]) \cap C^2((-1, 1))$ and parts of the theorem are stated in the following, where we adopted the notation to our setting. Moreover, higher regularity solutions are provided in [1, Lemma 4.1.3]. Let us denote $f \in L^2_{loc}(0,\infty; X)$ with X being a Banach space, iff $f \in L^2(0, T; X)$ for every $T > 0$. The uniformly local variant is denoted by $L^2_{uloc}([0,\infty); X)$ and comprises all strongly measurable functions $f : [0,\infty) \to X$ with $\|f\|_{L^2_{uloc}(0,\infty;X)} = \sup\limits_{t\geq 0}\|f\|_{L^2(t,t+1;X)} < \infty$.

Theorem 4.10. (Well-posedness of the state equation, [1, Theorem 4.1.1]). *Let $\mathcal{B}u \in L^2(0,\infty; H^1_n(\Omega))\cap L^\infty(0,\infty; H_{div}(\Omega))$. Then for every $\varphi_0 \in H^1_{(0)}(\Omega)$ with $\mathcal{E}_0 < \infty$ there exists a unique solution $\varphi \in BC([0,\infty); H^1_{(0)}(\Omega))$ of* (4.64) *with $\varphi_t \in L^2(0,\infty; H^{-1}_{(0)}(\Omega))$, $\mu \in L^2_{uloc}([0,\infty); H^1(\Omega))$. This solution satisfies*

$$\|\varphi\|^2_{L^\infty(0,\infty;H^1(\Omega))} + \|\varphi_t\|^2_{L^2(0,\infty;H^{-1}_{(0)}(\Omega))} + \|\nabla\mu\|^2_{L^2(0,\infty;\Omega)} \leq C\left(\mathcal{E}_0 + \|\mathcal{B}u\|^2_{L^2(0,\infty;\Omega)}\right), \tag{4.69}$$

and

$$\|\varphi\|^2_{L^2_{uloc}([0,\infty);W^{2,p}(\Omega))} + \|\mathcal{F}'(\varphi)\|^2_{L^2_{uloc}([0,\infty);L^p(\Omega))} \leq C_p\left(\mathcal{E}_0 + \|\mathcal{B}u\|^2_{L^2(0,\infty;\Omega)}\right) \tag{4.70}$$

where $p = 6$ if $d = 3$ and $p \in (1,\infty)$ if $d = 2$. Here C, C_p are independent of $\mathcal{B}u$ and φ_0. Moreover, for every $R, T > 0$ the solution

$$\varphi \in Y := L^2(0, T; W^{2,p}(\Omega)) \cap H^1(0, T; H^{-1}_{(0)}(\Omega))$$

depends continuously on

$$(\varphi_0, \mathcal{B}u) \in X := H^1(\Omega) \times L^1(0, T; H_{div}(\Omega)) \quad \textit{such that} \quad \mathcal{E}_0 + \|\mathcal{B}u\|_{L^2(0,\infty;H^1(\Omega))} \leq R$$

with respect to the weak topology on Y and the strong topology on X.

Due to Theorem 4.10, the optimal control problem (4.63)-(4.64)-(4.65) can be expressed as a reduced optimal control problem given by

$$\min_{u\in U_{ad}} \hat{J}(u) := J(\varphi(u), u) \quad \text{s.t. } (\varphi(u), u) \text{ satisfies (4.64) with (4.66), (4.67).} \tag{4.71}$$

Theorem 4.11. (Existence of an optimal control). *Problem* (4.71) *admits a solution $\bar{u} \in U_{ad}$.*

Proof. Step 1: Boundedness of the state: Due to the energy estimates (4.69) and (4.70) it holds

$$\|\varphi\|^2_{L^2(0,T;H^2(\Omega))} + \|\varphi_t\|^2_{L^2(0,T;H^{-1}_{(0)}(\Omega))} \leq c\left(E_0 + \|\mathcal{B}u\|^2_{L^2(0,T;\Omega)}\right),$$

such that we can find an upper bound $M > 0$, since $\mathcal{B}$ is linear and bounded.
Step 2: Find a candidate for an optimal control: Since the cost functional is bounded from below ($\hat{J} \geq 0$) and $U_{ad} \neq \emptyset$, the infimum

$$J^* := \inf_{u\in U_{ad}} \hat{J}(u) = \inf_{u\in U_{ad}} J(\varphi(u), u)$$

exists. Let $\{u_n\}_{n\in\mathbb{N}} \subset U_{ad}$ be a minimizing sequence, i.e.

$$\lim_{n\to\infty} J(\varphi(u_n), u_n) = J^*.$$

Since U_{ad} is closed, convex and bounded in $L^2(0,T;\mathbb{R}^m) \supset L^\infty(0,T;\mathbb{R}^m)$, it is weakly sequentially compact according to Theorem 2.38. As a result, we can extract a subsequence (denoted by the same name), which converges weakly in $L^\infty(0,T;\mathbb{R}^m)$ to some $\bar{u} \in U_{ad}$. Weak convergence $\mathcal{B}u_n \rightharpoonup \mathcal{B}\bar{u}$ in $L^2(0,T;H^1(\Omega))$ follows from the linearity and boundedness of the operator $\mathcal{B}$.

Step 3: Define associated sequence of states: Let $\{\varphi_n\}_{n\in\mathbb{N}}$ denote the corresponding sequence of states, i.e. $\varphi_n := \varphi(u_n)$ and let $\{\mu_n\}_{n\in\mathbb{N}}$ denote the corresponding sequence of chemical potentials. From Theorem 4.10 it follows that $\varphi_n \in \tilde{Y} := W(0,T;H^1_{(0)}(\Omega)) \cap L^2(0,T;H^2(\Omega))$. Since $\tilde{Y}$ is reflexive, there exists a subsequence (denoted by the same name) that converges weakly to some $\bar{\varphi} \in \tilde{Y}$ according to Theorem 2.38. Analogously, we obtain weak convergence for the chemical potential to some $\bar{\mu} \in L^2(0,T;H^1(\Omega))$.

Step 4: Show $(\bar{\varphi}, \bar{u})$ is admissible: It remains to show that the pair $(\bar{\varphi}, \bar{u})$ is admissible, i.e. $\bar{\varphi} = \varphi(\bar{u})$. We consider

$$\langle \varphi_{n,t}(t), v\rangle_{H^{-1}(\Omega),H^1_0(\Omega)} + ((\mathcal{B}u_n(t))\cdot\nabla\varphi_n(t), v)_{L^2(\Omega)} + b(\nabla\mu_n(t),\nabla v)_{L^2(\Omega)} = 0, \tag{4.72a}$$

$$\sigma\varepsilon(\nabla\varphi_n(t),\nabla v)_{L^2(\Omega)} + \frac{\sigma}{\varepsilon}(\mathcal{F}'(\varphi_n(t)), v)_{L^2(\Omega)} = (\mu_n(t), v)_{L^2(\Omega)}, \tag{4.72b}$$

for all $v \in H^1(\Omega)$ and for almost all $t \in [0,T]$. While passing to the limit $n \to \infty$ in the weak formulation is clear for the linear terms, the nonlinear ones require further investigation. Since $W(0,T) \cap L^2(0,T;H^2(\Omega))$ is compactly embedded in $L^2(0,T;H^1(\Omega))$ (see [164, Section 8, Corollary 4]), the sequence $\{\varphi_n\}_{n\in\mathbb{N}}$ converges strongly to $\bar{\varphi}$ in $L^2(0,T;H^1(\Omega))$. For the control term we have for $v \in H^1(\Omega)$ the splitting

$$\begin{aligned}\int_0^T (\mathcal{B}u_n(t)\cdot\nabla\varphi_n(t) - \mathcal{B}\bar{u}(t)\cdot\nabla\bar{\varphi}(t), v)_{L^2(\Omega)}\,dt = \\ \int_0^T (\mathcal{B}u_n(t)\cdot\nabla(\varphi_n(t)-\bar{\varphi}(t)), v)_{L^2(\Omega)}\,dt + \int_0^T (\mathcal{B}(u_n(t)-\bar{u}(t))\cdot\nabla\bar{\varphi}(t), v)_{L^2(\Omega)}\,dt.\end{aligned}$$

Due to $\bar{\varphi} \in L^2(0,T;H^2(\Omega))$ it holds $\nabla\bar{\varphi} \in L^2(0,T;H^1(\Omega))$ and the product $v\cdot\nabla\bar{\varphi} \in L^2(0,T;L^2(\Omega))$ gives rise to a continuous linear functional on $L^2(0,T;H^1(\Omega))$. Hence, the right term vanishes for $n \to \infty$ by definition of weak convergence. For the left term we estimate

$$\left|\int_0^T (\mathcal{B}u_n(t)\cdot\nabla(\varphi_n(t)-\bar{\varphi}(t)), v)_{L^2(\Omega)}\,dt\right| \le \|\mathcal{B}u_n\|_{L^2(0,T;H^1(\Omega))}\|\varphi_n - \bar{\varphi}\|_{L^2(0,T;H^1(\Omega))}\|v\|_{H^1(\Omega)},$$

which vanishes for $n \to \infty$ due to the convergence of $\{\varphi_n\}_{n\in\mathbb{N}}$. For the nonlinearity $\mathcal{F}'_{pol}$ we use the inequality from [101, Theorem 1.45]

$$|b^3 - a^3| \le 3(|a|^2 + |b|^2)|b - a|$$

for $a, b \in \mathbb{R}$ which leads to

$$|\mathcal{F}'_{pol}(\varphi) - \mathcal{F}'_{pol}(\psi)| = |\varphi^3 - \varphi - \psi^3 + \psi| \le 3(|\varphi|^2 + |\psi|^2)\,|\varphi - \psi| + |\varphi - \psi|$$

for all $\varphi, \psi \in \mathbb{R}$. As a consequence, we obtain the estimate

$$\begin{aligned}\left|\int_0^T (\mathcal{F}'_{pol}(\varphi_n(t)) - \mathcal{F}'_{pol}(\bar{\varphi}(t)), v)_{L^2(\Omega)}\,dt\right| \le \\ 3(\|\varphi_n^2\|_{L^2(0,T;L^2(\Omega))} + \|\bar{\varphi}^2\|_{L^2(0,T;L^2(\Omega))} + 1)\|\varphi_n - \bar{\varphi}\|_{L^2(0,T;L^2(\Omega))}\|v\|_{H^1(\Omega)},\end{aligned}$$

which gives the desired convergence due to the Sobolev imbedding $L^\infty(0,T;H^1(\Omega)) \subset L^4(0,T;L^4(\Omega))$.

Step 5: Optimality of $\bar{u}$: Since $\hat{J}$ is a continuous, convex functional we obtain from Theorem 2.39 that $\hat{J}$ is weakly lower semicontinuous, i.e.

$$u_n \rightharpoonup \bar{u} \quad \Rightarrow \quad \liminf_{n\to\infty} \hat{J}(u_n) \geq \hat{J}(\bar{u}).$$

Therefore, it follows $\hat{J}(\bar{u}) = \inf_{u\in U_{ad}} \hat{J}(u)$ and $\bar{u}$ is an optimal control. □

The optimal control problem (4.71) is a non-convex programming problem so that different minima might exist. Numerical solution methods will converge in a local minimum which is close to the initial point. In order to compute a locally optimal solution to (4.71), we consider the first-order necessary optimality condition given by the variational inequality

$$\langle \hat{J}'(\bar{u}), u - \bar{u}\rangle_{U^*,U} \geq 0 \quad \forall u \in U_{ad}. \tag{4.73}$$

Following the standard Lagrange technique from Section 2.3, we derive that (4.73) is equivalent to

$$\int_0^T \sum_{i=1}^m \left(\gamma \bar{u}_i(t) + \int_\Omega (\chi_i(x)\cdot\nabla\varphi(t,x))\bar{p}(t,x)dx\right)(u_i(t)-\bar{u}_i(t))dt \geq 0 \tag{4.74}$$

for all $u \in U_{ad}$ where the function $\bar{p}$ is a solution to the adjoint equations

$$\begin{aligned} -p_t - \sigma\varepsilon\Delta q + \frac{\sigma}{\varepsilon}\mathcal{F}''(\bar{\varphi})q - \mathcal{B}u\cdot\nabla p &= -\beta_1(\bar{\varphi}-\varphi_d) && \text{in } (0,T]\times\Omega, && (4.75a)\\ -q - b\Delta p &= 0 && \text{in } (0,T]\times\Omega, && (4.75b)\\ \partial_n p = \partial_n q &= 0 && \text{on } [0,T]\times\partial\Omega, && (4.75c)\\ p(T,\cdot) &= -\beta_2(\bar{\varphi}(T,\cdot)-\varphi_T) && \text{in } \Omega. && (4.75d) \end{aligned}$$

The variable $\bar{\varphi}$ in (4.75) denotes the solution to state equation (4.64) associated with an optimal control $\bar{u}$. The weak form of the adjoint system is given as follows: find the adjoint states p and q such that $p(T,\cdot) = -\beta_2(\bar{\varphi}(T,\cdot)-\varphi_T)$ and

$$\begin{aligned} -\langle p_t(t), v\rangle_{H^{-1}(\Omega),H_0^1(\Omega)} + (\nabla q(t),\nabla v)_{L^2(\Omega)} + \frac{\sigma}{\varepsilon}(\mathcal{F}''(\bar{\varphi}(t))q(t),v)_{L^2(\Omega)} & && (4.76a)\\ -(\mathcal{B}u(t)\cdot\nabla p(t),v)_{L^2(\Omega)} &= (\bar{\varphi}(t)-\varphi_d(t),v)_{L^2(\Omega)}, \\ -(q(t),v)_{L^2(\Omega)} + b(\nabla p,\nabla v)_{L^2(\Omega)} &= (\mu(t),v)_{L^2(\Omega)}, && (4.76b) \end{aligned}$$

holds for all $v \in H^1(\Omega)$ and almost all $t \in (0,T)$.

Discretization

For the discretization, we first discretize the state equation in time using the convex-concave splitting in (2.26) leading to an unconditional energy stable scheme. For the spatial discretization of the state equation, we use h-adaptive finite elements according to Algorithm 1 where the estimator is based on the gradient jump, compare (4.39). In order to make computations efficient, we only refine once at each time instance and introduce an additional coarsening strategy according to [114, Section 8.3]. For the discretization of the adjoint state, we follow the *first-discretize-then-optimize* concept (compare Remark 2.44). In particular, we note that for simplicity we use the same adapted meshes for the state as for the adjoint variable in order to circumvent the handling of a two-mesh system. In our numerical example we observe that the state and adjoint variable are related such that a refinement for the state is also suitable for the adjoint state. It could also be of interest to investigate further adaptivity concepts for the adjoint state, especially if the solution properties for the state and adjoint state differ strongly.

4.7.2 Numerical results

Note that this example is published in [87].

Problem setting

The choices for the problem data and parameters are listed in Table 9.

	name	value
Ω	spatial domain	$(0,1) \times (0,1) \subset \mathbb{R}^2$
T	end time	0.0125
Δt	time step size	$2.5 \cdot 10^{-05}$
b	constant mobility	$2 \cdot 10^{-05}$
ε	interface parameter	0.02
σ	surface tension	25.98
$\mathcal{F}$	nonlinear free energy	$\mathcal{F}_{pol}$ (2.22)
φ_0	initial condition	circle with radius 0.35 and center $(0.5, 0.5)$
γ	regularization parameter	0.0001
$m = 1$, χ_1	control shape	$\chi_1(x) = (\sin(\pi x_1)\cos(\pi x_2), -\sin(\pi x_2)\cos(\pi x_1))^T$

Table 9: Problem data and parameters of the optimal control problem (4.71)

In order to fulfill the Courant–Friedrichs–Lewy (CFL) type condition, we impose the control constraints $u_a = 0, u_b = 50$ and demand $h_{\min} > 0.00177$. In the cost functional we use $\gamma = 0.0001$, $\beta_1 = 20$ and $\beta_2 = 20$. The desired state is shown in Figure 52. The initial state φ_0 coincides with $\varphi_d(0)$.
The optimization is initialized with an input control $u = 0 \in U_{ad}$. We compute the POD basis with respect to the $L^2(\Omega)$-inner product for the snapshot ensemble formed by the desired phase field φ_d, which is discretized using adaptive finite elements. For the solution of the optimal control problem, we use a projected gradient method. Figure 53 shows the finite element solution and the POD-DEIM solution for the phase field using $\ell = 10$ and $\ell = 20$ POD modes, respectively. It turns out that a large number of POD modes is needed in order to smoothen out oscillations due to the convection.

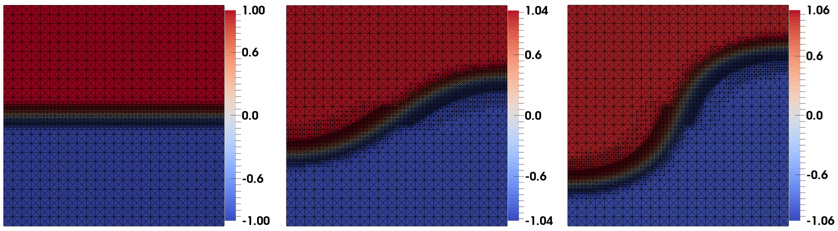

Figure 52: Desired phase field at t_0, t_{250}, t_{500} with adaptive meshes

Table 10 (left) summarizes the iteration history for the finite element projected gradient method where we used the stopping criterion $\|\hat{J}'(u^k)\| < 0.01 \cdot \|\hat{J}'(u^0)\| + 0.01$. Table 10 (right) tabulates the POD-DEIM optimization. Note that the value of the POD cost functional $\hat{J}_\ell(u^k)$ stagnates due to the POD error. The value of the full-order cost functional at the POD solution is $\hat{J}(\bar{u}_{\text{POD}}) = 7.31 \cdot 10^{-04}$. If $\ell = 20$ POD modes are used, the relative $L^2(0,T;\Omega)$-error between the finite element and the POD-DEIM optimal solution is for the phase field $\text{err}_\varphi = 7.38 \cdot 10^{-03}$. In Table 11 the computational times for the uniform finite element, adaptive finite element, POD and POD-DEIM optimization are listed. The offline costs for POD when using spatially adapted snapshots are as follows: the interpolation of the snapshots takes 212 seconds, the POD basis

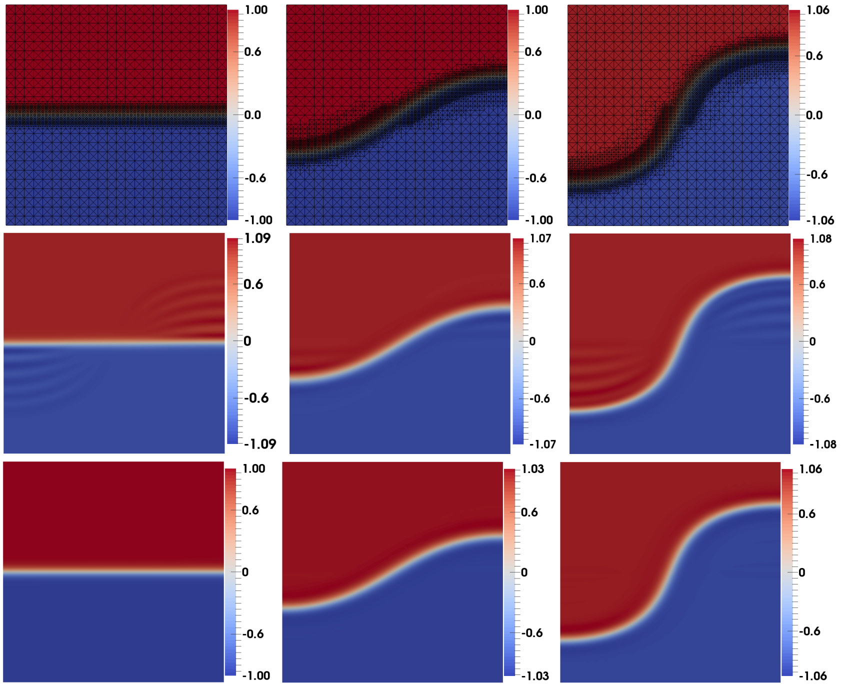

Figure 53: Finite element (top) and POD-DEIM reduced-order solution with $\ell = 10$ (middle) and $\ell = 20$ (bottom) of the phase field φ at $t = t_0, t_{250}, t_{500}$ with adaptive meshes

k	$\hat{J}(u^k)$	$\lVert\hat{J}'(u^k)\rVert$	s_k
0	$8.61 \cdot 10^{0}$	$2.85 \cdot 10^{0}$	1.0
1	$6.48 \cdot 10^{-01}$	$2.32 \cdot 10^{0}$	0.25
2	$1.90 \cdot 10^{-02}$	$4.56 \cdot 10^{-01}$	0.25
3	$3.82 \cdot 10^{-03}$	$1.93 \cdot 10^{-01}$	0.25
4	$1.18 \cdot 10^{-03}$	$8.45 \cdot 10^{-02}$	0.25
5	$6.80 \cdot 10^{-04}$	$3.67 \cdot 10^{-02}$	

k	$\hat{J}_\ell(u^k)$	$\lVert\hat{J}_\ell'(u^k)\rVert_{U_h}$	s_k
0	$8.77 \cdot 10^{0}$	$2.81 \cdot 10^{0}$	1.0
1	$7.98 \cdot 10^{-01}$	$2.41 \cdot 10^{0}$	0.25
2	$5.79 \cdot 10^{-02}$	$3.67 \cdot 10^{-01}$	0.25
3	$5.02 \cdot 10^{-02}$	$1.63 \cdot 10^{-01}$	0.25
4	$4.76 \cdot 10^{-02}$	$7.45 \cdot 10^{-02}$	0.25
5	$4.76 \cdot 10^{-02}$	$3.48 \cdot 10^{-02}$	

Table 10: Iteration history of the finite element optimization (left) and the POD-DEIM optimization (right) with $\ell = 20$. The Armijo step size is denoted by s_k

	uniform FE	adaptive FE	POD	POD-DEIM
optimization	36868	5805	675	0.3
→ solve each state eq.	1660	348	42	0.02
→ solve each adjoint eq.	761	121	16	0.01

Table 11: Computational times (sec) for the finite element, POD and POD-DEIM optimization

computation costs 40 seconds and the computations of a DEIM basis takes 30 seconds. In comparison, the use of uniformly discretized snapshots leads to the computational time of 243 seconds for POD basis computation and 193 seconds for the DEIM computations.

The numerical tests show that the use of POD-DEIM reduced-order modeling combined with spatially adapted snapshots leads to large speed up factors compared with a high-fidelity finite element optimization.

5 Summary, conclusions and outlook

This thesis is concerned with offline adaptivity strategies in model order reduction using proper orthogonal decomposition (POD) for time-dependent systems and optimal control problems. In order to construct suitable snapshots which identify key properties of the solution data, we propose two major strategies.

In the first part, we consider a snapshot location strategy for model order reduction in optimal control. The selection of suitable time instances for snapshot generation is based on a reformulation of the optimality system into a biharmonic equation and uses residual based a-posteriori error estimates in order to compute a time-adaptive grid. We propose adaptive strategies which are tailored for the state and adjoint state / control and ensure efficiency by a heuristic observation in the numerical experiments in which spatial and temporal discretization decouple. This allows to use a coarse spatial grid in the offline phase. At the same time, we compute an approximation of the optimal control which can be used as input control for snapshot generation. We further address the case of located control and control constraints. An error analysis for the error between the true (unknown) optimal solution and the POD reduced-order solution is carried out. It reveals that a time-adaptive grid which is suitable for the state might not be a suitable time grid for the adjoint state / control and vice versa. In numerical experiments we consider time-dependent systems which develop steep transitions within the temporal domain. We demonstrate that the use of a time-adaptive grid leads to a good approximation quality of the resulting reduced-order model with only a little number of time instances and is thus efficient. Further, we observe that the adaptive snapshot grid is also a suitable time grid for the reduced-order optimization, since we already detect the major dynamical structures related to the optimal solution. Hence, the proposed methodology is not only an offline adaptive strategy, but also provides benefits for the online solution.
As an outlook, we sketch ideas for future research questions in Section 3.9 comprising the case of a nonlinear state equation and the application of the concept to a model predictive control problem. Moreover, different variations of the problem setting like the inclusion of a transport term or the consideration of state constraints could be addressed. Regarding the numerical experiments it would be of interest to carry out computations for a 2d or 3d setting with complex dynamical structures in time, where we expect the proposed snapshot location strategy and time discretization to be advantageous.

In the second part, we include spatial adaptivity in the offline phase for snapshot generation in model order reduction with POD. The goal is to identify dominant spatial structures by means of adaptive finite element methods. This enables a fast snapshot sampling in comparison to the use of a very fine uniform discretization while a prescribed accuracy of the finite element solution is ensured by the use of residual based error indicators. The major challenge in the context of POD model order reduction is that space-adapted snapshots have different numbers and locations of discretization points which means in a fully discrete setting that the snapshots are vectors of different lengths. In order to tackle this issue, we consider the POD method from an infinite-dimensional perspective and compute an eigenvalue problem for the snapshot gramian which is set up explicitly. It is an advantage of this approach that it allows snapshot discretizations stemming from r-, p- and h-adaptive schemes. In case of space-adapted snapshots with non-nested meshes, the computation of the snapshot gramian involves the integration over cut finite elements. For the numerical implementation, we propose a strategy which is based on an integration over the boundary of the cut polyhedra utilizing Stoke's formula. This is numerically intensive, however a speed up can be achieved using parallelization. An error analysis for the error between the true (unknown) solution and the POD reduced-order solution using space-adapted snapshots is carried out and reveals a further dependency on the spatial discretization error in comparison to existing error estimates. In numerical examples comprising a linear heat equation and a nonlinear Cahn–Hilliard system, the combination of POD with space-adapted snapshots in case of hierarchical, nested meshes and in case of non-nested meshes is presented. We obtain large speed up factors in the offline times for space-adapted finite element solutions in comparison with a very fine uniform discretization. Moreover, the POD reduced-order online simulation time is very fast in comparison to the full-order model. At the same time we can achieve a good accuracy. In the context of incompressible flows governed by the unsteady Navier–Stokes equations, the challenge arises that due to the use of h-adapted finite elements a weakly divergence-free property only holds in the current finite element space (and not in a common sense). Hence, the POD modes do not automatically inherit a divergence-free property like in the static case and the reduced-order model is not stable. We propose two solution concepts in order to derive a stable reduced-order model. The first approach is based on a velocity reduced-order modeling and utilizes a projection of the snapshots or the POD modes such that the projected data is weakly divergence-free in a common sense. It is an advantage of this approach that the reduced-order model is cheap to solve since it only depends on the velocity. The second approach is a velocity-pressure reduced-order model, where the stability is guaranteed by enriching the reduced velocity space with pressure supremizer functions. In this way, it is shown that the inf-sup stability of the finite element model carries over to an inf-sup stability of the reduced-order model. The advantage of this approach is that the model delivers not only a velocity but also a pressure approximation, which is of interest in many applications. The numerical experiment reveals that both approaches deliver equally accurate reduced-order solutions.

Finally, we utilize h-adaptive finite element snapshots for the solution of an optimal control problem subject to a Cahn–Hilliard system, where the control enters the system in the transport term. In order to adapt the reduced-order model accuracy within the course of the optimization, in [5, 6] a trust-region POD framework with space-adapted snapshots is used for a similar optimal control problem. It is of future research interest to apply the developed strategies to the simulation and optimal control of a coupled Cahn–Hilliard Navier–Stokes system describing two-phase flow.

6 Appendix

6.1 Snapshot location: uniform time mesh misses a signal

We construct an example in the following way. The optimal state is chosen as

$$y(t, x) = \sin(\pi x) \cdot g(t),$$

where g is the probability density function of the normal distribution given as

$$g(t) = \frac{1}{\sqrt{2\pi\varepsilon}} \exp\left(-\frac{(t-0.3)^2}{2\varepsilon}\right)$$

with variance $\varepsilon = 10^{-04}$. The true optimal solution, its contour lines and the density function g are shown in Section 6.1.

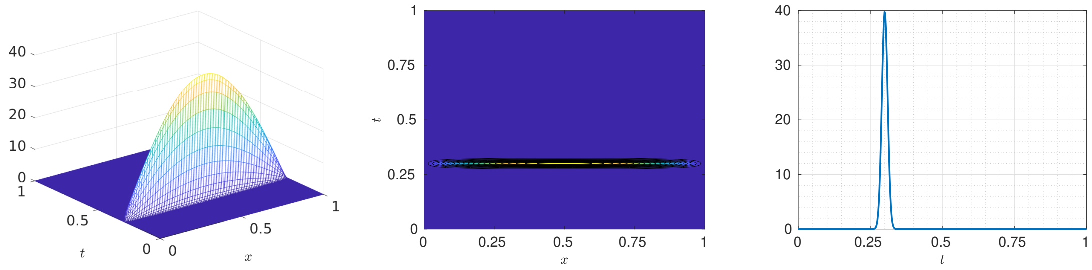

Figure 54: Optimal state (left), its contour lines (middle) and the density function g (right)

We choose the source function $f(t, x) = 0$ and construct the optimal control as

$$u(t, x) = \sin(\pi x)(g_t(t) + \pi^2 g(t)).$$

The corresponding desired state is given by

$$y_d(t, x) = \sin(\pi x)((1 + \pi^4)g(t) - g_{tt}(t)).$$

This academic example is constructed such that it shows a scenario in which a uniform discretization misses a dynamical structure. Thus, we choose a uniform time grid with $n = 5$, i.e. $\Delta t = 1/n = 0.2$. With this choice, it holds true $y(t_j, x) < 10^{-15}, p(t_j, x) < 10^{-15}$ for $j = 0, \ldots, n+1$ and all $x \in [0, 1]$. Therefore, both state and adjoint snapshot sets will only contain zero vectors, i.e. a POD computes an arbitrary orthonormal basis and the resulting POD state and adjoint solution will be zero. This reflects the snapshots correctly, however it does not reflect the true solution at all. In particular, it misses the time-dependent signal at $t = 0.3$ determined by the normal distribution g. In contrary, the time-adaptive algorithm is able to detect this signal with the same number of time instances. The space-time grid is shown in Section 6.1. The resulting POD state solution is shown in the same figure (middle, right). We observe that the time peak at 0.3 is detected. In comparison to the true solution, the scaling is still not good, but this is due to the temporal resolution and can be improved by enlarging the number of time instances. We see that the dynamic in the spatial domain which is determined by the sin-function is well captured. This is in contrast to the uniform time discretization, in which an arbitrary (POD) orthonormal basis will be chosen.

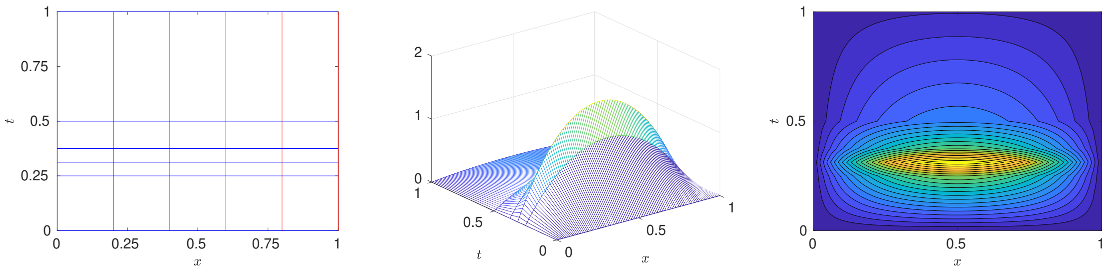

Figure 55: Space-time grid with $n = 5$ and $h_+ = 1/5$ (left), POD optimal state (middle) and its contour lines (right)

References

[1] H. Abels. Diffuse interface models for two-phase flows of viscous incompressible fluids. *Lecture Notes, Max Planck Institute for Mathematics in the Sciences*, (36), 2007.

[2] H. Abels, H. Garcke, and G. Grün. Thermodynamically consistent, frame indifferent diffuse interface models for incompressible two-phase flows with different densities. *Mathematical Models and Methods in Applied Sciences*, 22(3), 2012.

[3] K. Afanasiev and M. Hinze. Adaptive control of a wake flow using proper orthogonal decomposition. *Lecture Notes in Pure and Applied Mathematics*, pages 317–332, 2001.

[4] M. Ainsworth and J. T. Oden. *A posteriori error estimation in finite element analysis*, volume 37. John Wiley & Sons, 2011.

[5] J. O. Alff. Trust Region POD for Optimal Control of Cahn-Hilliard Systems. Master's thesis, Universität Hamburg, 2018.

[6] J. O. Alff, C. Gräßle, and M. Hinze. Adaptive trust-region POD for optimal control of the Cahn-Hilliard equation. *PAMM*, 18(1):e201800453, 2018.

[7] M. Ali, K. Steih, and K. Urban. Reduced basis methods with adaptive snapshot computations. *Advances in Computational Mathematics*, 43(2):257–294, 2017.

[8] A. Alla and M. Falcone. A time-adaptive POD method for optimal control problems. *IFAC Proceedings Volumes*, 46(26):245–250, 2013.

[9] A. Alla, C. Gräßle, and M. Hinze. A residual based snapshot location strategy for POD in distributed optimal control of linear parabolic equations. *IFAC-PapersOnLine*, 49(8):13–18, 2016.

[10] A. Alla, C. Gräßle, and M. Hinze. Snapshot location for POD in control of a linear heat equation. *PAMM*, 16(1):781–782, 2016.

[11] A. Alla, C. Gräßle, and M. Hinze. A-posteriori snapshot location for POD in optimal control of linear parabolic equations. *ESAIM: M2AN*, 52(5):1847–1873, 2018.

[12] A. Alla and J. N. Kutz. Nonlinear model order reduction via dynamic mode decomposition. *SIAM Journal on Scientific Computing*, 39(5):B778–B796, 2017.

[13] M. S. Alnæs, J. Blechta, J. Hake, A. Johansson, B. Kehlet, A. Logg, C. Richardson, J. Ring, M. E. Rognes, and G. N. Wells. The FEniCS project version 1.5. *Archive of Numerical Software*, 3(100):9–23, 2015.

[14] R. Altmann and J. Heiland. Regularization and Rothe discretization of semi-explicit operator DAEs. *International Journal of Numerical Analysis and Modeling*, 15(3):452–478, 2018.

[15] C. A. Andrews, J. M. Davies, and G. R. Schwarz. Adaptive data compression. *Proceedings of the IEEE*, 55(3):267–277, 1967.

[16] A. C. Antoulas. *Approximation of large-scale dynamical systems*, volume 6. Siam, 2005.

[17] E. Arian, M. Fahl, and E. W. Sachs. Trust-region proper orthogonal decomposition for flow control. Technical report, 2000.

[18] P. Astrid, S. Weiland, K. Willcox, and T. Backx. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control*, 53(10):2237–2251, 2008.

[19] I. Babuška. Error-bounds for finite element method. *Numerische Mathematik*, 16(4):322–333, 1971.

[20] I. Babuska, B. A. Szabo, and I. N. Katz. The p-version of the finite element method. *SIAM Journal on Numerical Analysis*, 18(3):515–545, 1981.

[21] A. Bailey, W. C. K. Poon, R. J. Christianson, A. B. Schofield, U. Gasser, V. Prasad, S. Manley, P. N. Segre, L. Cipelletti, W. V. Meyer, et al. Spinodal decomposition in a model colloid-polymer mixture in microgravity. *Physical review letters*, 99(20):205701, 2007.

[22] F. Ballarin, A. Manzoni, A. Quarteroni, and G. Rozza. Supremizer stabilization of POD–Galerkin approximation of parametrized steady incompressible Navier–Stokes equations. *International Journal for Numerical Methods in Engineering*, 102(5):1136–1161, 2015.

[23] L. Baňas, A. Novick-Cohen, and R. Nürnberg. The degenerate and non-degenerate deep quench obstacle problem: a numerical comparison. *Networks and Heterogeneous Media*, 8:37–64, 2013.

[24] L. Baňas and R. Nürnberg. Numerical approximation of a non-smooth phase-field model for multicomponent incompressible flow. *ESAIM: Mathematical Modelling and Numerical Analysis*, 51(3):1089–1117, 2017.

[25] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera. An ‘empirical interpolation’ method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique*, 339(9):667–672, 2004.

[26] M. Baumann, P. Benner, and J. Heiland. Space-Time Galerkin POD with Application in Optimal Control of Semilinear Partial Differential Equations. *SIAM Journal on Scientific Computing*, 40(3):A1611–A1641, 2018.

[27] C. A. Beattie and S. Gugercin. Model reduction by rational interpolation. *Model Reduction and Algorithms: Theory and Applications, P. Benner, A. Cohen, M. Ohlberger, and K. Willcox, eds., Computational Science & Engineering*, 15:297–334, 2014.

[28] P. Benner and R. Rannacher. Introduction to part III, Adaptivity and Model Reduction, in Trends in PDE constrained optimization. *International Series of Numerical Mathematics*, 165, 2014.

[29] G. Berkooz, P. Holmes, and J. L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual review of fluid mechanics*, 25(1):539–575, 1993.

[30] C. Bernardi and E. Süli. Time and space adaptivity for the second-order wave equation. *Mathematical Models and Methods in Applied Sciences*, 15(02):199–225, 2005.

[31] A. Bertram and R. Zimmermann. Theoretical investigations of the new cokriging method for variable-fidelity surrogate modeling: Well-posedness and maximum likelihood training. *Advances in Computational Mathematics*, 44(6):1693–1716, 2018.

[32] M. Besier and W. Wollner. On the pressure approximation in nonstationary incompressible flow simulations on dynamically varying spatial meshes. *International Journal for Numerical Methods in Fluids*, 69(6):1045–1064, 2012.

[33] L. Blank, M. Butz, and H. Garcke. Solving the Cahn-Hilliard variational inequality with a semi-smooth Newton method. *ESAIM: Control, Optimisation and Calculus of Variations*, 17(4):931–954, 2011.

[34] L. Blank, H. Garcke, M. Hassan Farshbaf-Shaker, and V. Styles. Relating phase field and sharp interface approaches to structural topology optimization. *ESAIM: Control, Optimisation and Calculus of Variations*, 20(4):1025–1058, 2014.

[35] J. F. Blowey and C. M. Elliott. The Cahn–Hilliard gradient theory for phase separation with non-smooth free energy Part I: Mathematical analysis. *European Journal of Applied Mathematics*, 2(3):233–280, 1991.

[36] J. F. Blowey and C. M. Elliott. The Cahn–Hilliard gradient theory for phase separation with non-smooth free energy Part II: Numerical analysis. *European Journal of Applied Mathematics*, 3(2):147–179, 1992.

[37] D. Braess. *Finite elements: Theory, fast solvers, and applications in solid mechanics.* Cambridge University Press, 2007.

[38] M. Brand. Incremental singular value decomposition of uncertain data with missing values. In *European Conference on Computer Vision*, pages 707–720. Springer, 2002.

[39] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Mathematics of computation*, 31(138):333–390, 1977.

[40] F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *Publications mathématiques et informatique de Rennes*, (S4):1–26, 1974.

[41] J. Burkardt, M. Gunzburger, and H.-C. Lee. POD and CVT-based reduced-order modeling of Navier–Stokes flows. *Computer Methods in Applied Mechanics and Engineering*, 196(1-3):337–355, 2006.

[42] G. Büttner. Ein Mehrgitterverfahren zur optimalen Steuerung parabolischer Probleme. *Dissertation, Technische Universität Berlin*, 2004.

[43] J. W. Cahn and J. E. Hilliard. Free energy of a nonuniform system. I. Interfacial free energy. *The Journal of chemical physics*, 28(2):258–267, 1958.

[44] I. Capuzzo Dolcetta, S. Finzi Vita, and R. March. Area-preserving curve-shortening flows: from phase separation to image processing. *Interfaces and Free Boundaries*, 4(4):325–343, 2002.

[45] K. Carlberg. Adaptive h-refinement for reduced-order models. *International Journal for Numerical Methods in Engineering*, 102(5):1192–1210, 2015.

[46] J. Carr, M. E. Gurtin, and M. Slemrod. Structured phase transitions on a finite interval. *Archive for rational mechanics and analysis*, 86(4):317–351, 1984.

[47] C. Carstensen and R. Verfürth. Edge residuals dominate a posteriori error estimates for low order finite element methods. *SIAM Journal on Numerical Analysis*, 36(5):1571–1587, 1999.

[48] E. Casas. Pontryagin's principle for state-constrained boundary control problems of semilinear parabolic equations. *SIAM Journal on Control and Optimization*, 35(4):1297–1327, 1997.

[49] J. M. Cascon, C. Kreuzer, R. H. Nochetto, and K. G. Siebert. Quasi-optimal convergence rate for an adaptive finite element method. *SIAM Journal on Numerical Analysis*, 46(5):2524–2550, 2008.

[50] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing*, 32(5):2737–2764, 2010.

[51] Y. Chen. Model order reduction for nonlinear systems. Master's thesis, Massachusetts Institute of Technology, 1999.

[52] Z. Chen and K.-H. Hoffmann. Numerical solutions of the optimal control problem governed by a phase field model. In *Estimation and Control of Distributed Parameter Systems*, pages 79–97. Springer, 1991.

[53] P. G. Ciarlet and J.-L. Lions. Handbook of numerical analysis. Vol. 1. *Amsterdam and New York, North-Holland, 1990, 658*, 1990.

[54] P. G. Ciarlet and J.-L. Lions. Handbook of numerical analysis. Vol. 2, Finite Element Methods (part 1). *Amsterdam and New York, North-Holland, 1990, 658*, 1991.

[55] P. G. Ciarlet and P.-A. Raviart. A mixed finite element method for the biharmonic equation. *Mathematical aspects of finite elements in partial differential equations*, pages 125–145, 1974.

[56] P. Constantin and C. Foias. *Navier-Stokes equations.* The University of Chicago Press, 1988.

[57] N. Cressie. The origins of kriging. *Mathematical geology*, 22(3):239–252, 1990.

[58] R. Dautray and J.-L. Lions. *Mathematical Analysis and Numerical Methods for Science and Technology, Volume 5 Evolution Problems I.* Springer, 1992.

[59] J. C. De los Reyes. *Numerical PDE-constrained optimization.* Springer, 2015.

[60] W. Dörfler. A convergent adaptive algorithm for Poisson's equation. *SIAM Journal on Numerical Analysis*, 33(3):1106–1124, 1996.

[61] Z. Drmac and S. Gugercin. A new selection operator for the discrete empirical interpolation method - Improved a priori error bound and extensions. *SIAM Journal on Scientific Computing*, 38(2):A631–A648, 2016.

[62] J. Du, F. Fang, C. C. Pain, I. Navon, J. Zhu, and D. A. Ham. POD reduced-order unstructured mesh modeling applied to 2D and 3D fluid flow. *Computers & Mathematics with Applications*, 65(3):362–379, 2013.

[63] G. Dziuk. *Theorie und Numerik Partieller Differentialgleichungen.* Walter de Gruyter, 2010.

[64] C. M. Elliott. The Cahn-Hilliard model for the kinetics of phase separation. In *Mathematical models for phase change problems*, pages 35–73. Springer, 1989.

[65] C. M. Elliott, D. A. French, and F. Milner. A second order splitting method for the Cahn-Hilliard equation. *Numerische Mathematik*, 54(5):575–590, 1989.

[66] C. M. Elliott and H. Garcke. On the Cahn–Hilliard equation with degenerate mobility. *Siam Journal on Mathematical Analysis*, 27(2):404–423, 1996.

[67] C. M. Elliott and Z. Songmu. On the Cahn-Hilliard equation. *Archive for Rational Mechanics and Analysis*, 96(4):339–357, 1986.

[68] C. M. Elliott and A. Stuart. The global dynamics of discrete semilinear parabolic equations. *SIAM Journal on Numerical Analysis*, 30(6):1622–1663, 1993.

[69] L. C. Evans. *Partial differential equations.* American Mathematical Society, Providence, R.I., 2010.

[70] D. J. Eyre. Unconditionally gradient stable time marching the Cahn-Hilliard equation. *MRS Online Proceedings Library Archive*, 529, 1998.

[71] D. J. Eyre. An unconditionally stable one-step scheme for gradient systems. *Technical report*, pages 1–15, 1998.

[72] F. Fang, C. Pain, I. Navon, M. Piggott, G. Gorman, P. Allison, and A. Goddard. Reduced-order modelling of an adaptive mesh ocean model. *International journal for numerical methods in fluids*, 59(8):827–851, 2009.

[73] H. Fareed, J. R. Singler, Y. Zhang, and J. Shen. Incremental proper orthogonal decomposition for PDE simulation data. *Computers & Mathematics with Applications*, 75(6):1942–1960, 2018.

[74] L. Feng, X. Zeng, C. Chiang, D. Zhou, and Q. Fang. Direct nonlinear order reduction with variational analysis. In *Design, Automation and Test in Europe Conference and Exhibition, Proceedings*, volume 2, pages 1316–1321. IEEE, 2004.

[75] T. Franz, R. Zimmermann, S. Görtz, and N. Karcher. Interpolation-based reduced-order modelling for steady transonic flows via manifold learning. *International Journal of Computational Fluid Dynamics*, 28(3-4):106–121, 2014.

[76] J. d. Frutos, V. John, and J. Novo. Projection methods for incompressible flow problems with WENO finite difference schemes. *Journal of Computational Physics*, 309:368–386, 2016.

[77] K. Fukunaga. *Introduction to statistical pattern recognition.* Academic Press, New York, 1972.

[78] H. Garcke, C. Hecht, M. Hinze, C. Kahle, and K. F. Lam. Shape optimization for surface functionals in Navier-Stokes flow using a phase field approach. *Interfaces and Free Boundaries*, 18(2):219–261, 2016.

[79] H. Garcke, B. Nestler, and B. Stoth. On anisotropic order parameter models for multi-phase systems and their sharp interface limits. *Physica D: Nonlinear Phenomena*, 115(1-2):87–108, 1998.

[80] D. Gay and W. H. Ray. Identification and control of linear distributed parameter systems through the use of experimentally determined singular functions. *IFAC Proceedings Volumes*, 20(1):173–179, 1987.

[81] M. S. Gockenbach. *Understanding and implementing the finite element method*, volume 97. SIAM, 2006.

[82] W. Gong, M. Hinze, and Z. Zhou. Space-time finite element approximation of parabolic optimal control problems. *Journal of Numerical Mathematics*, 20(2):111–146, 2012.

[83] C. Gräßle, M. Gubisch, S. Metzdorf, S. Rogg, and S. Volkwein. POD basis updates for nonlinear PDE control. *at - Automatisierungstechnik*, 65(5):298–307, 2017.

[84] C. Gräßle and M. Hinze. The combination of POD model reduction with adaptive finite element methods in the context of phase field models. *PAMM*, 17(1):47–50, 2017.

[85] C. Gräßle and M. Hinze. POD reduced-order modeling for evolution equations utilizing arbitrary finite element discretizations. *Advances in Computational Mathematics*, 44(6):1941–1978, 2018.

[86] C. Gräßle, M. Hinze, J. Lang, and S. Ullmann. POD model order reduction with space-adapted snapshots for incompressible flows. *Accepted for publication in Advances in Computational Mathematics, preprint available https://arxiv.org/abs/1810.03892*, 2019.

[87] C. Gräßle, M. Hinze, and N. Scharmacher. POD for optimal control of the Cahn-Hilliard system using spatially adapted snapshots. In *Numerical Mathematics and Advanced Applications ENUMATH 2017*, pages 703–711. Springer International Publishing, 2019.

[88] L. Grüne and J. Pannek. Nonlinear model predictive control. In *Nonlinear Model Predictive Control*, pages 43–66. Springer, 2011.

[89] M. Gubisch and S. Volkwein. Proper orthogonal decomposition for linear-quadratic optimal control. *In P. Benner, A. Cohen, M. Ohlberger, and K. Willcox, editors, Model Reduction and Approximation: Theory and Algorithms*, 15:5–66, 2017.

[90] F. Guillén-González and G. Tierra. On linear schemes for a Cahn–Hilliard diffuse interface model. *Journal of Computational Physics*, 234:140–171, 2013.

[91] J. Hadamard. Sur les problèmes aux dérivées partielles et leur signification physique. *Princeton University Bulletin*, XIII(4), 1902.

[92] F. Haußer, S. Rasche, and A. Voigt. The influence of electric fields on nanostructures - simulation and control. *Mathematics and Computers in Simulation*, 80(7):1449–1457, 2010.

[93] M. Heinkenschloss and E. Sachs. Numerical solution of a constrained control problem for a phase field model. In *Control and Estimation of Distributed Parameter Systems: Nonlinear Phenomena*, pages 171–187. Springer, 1994.

[94] J. Heywood and R. Rannacher. Finite Element Approximation of the Nonstationary Navier–Stokes Problem. I. Regularity of Solutions and Second-Order Error Estimates for Spatial Discretization. *SIAM Journal on Numerical Analysis*, 19(2):275–311, 1982.

[95] M. Hintermüller, M. Hinze, C. Kahle, and T. Keil. A goal-oriented dual-weighted adaptive finite element approach for the optimal control of a nonsmooth Cahn–Hilliard–Navier–Stokes system. *Optimization and Engineering*, 19(3):629–662, 2018.

[96] M. Hintermüller, M. Hinze, and M. H. Tber. An adaptive finite-element Moreau–Yosida-based solver for a non-smooth Cahn–Hilliard problem. *Optimization Methods and Software*, 26(4-5):777–811, 2011.

[97] M. Hintermüller and D. Wegner. Distributed optimal control of the Cahn–Hilliard system including the case of a double-obstacle homogeneous free energy density. *SIAM Journal on Control and Optimization*, 50(1):388–418, 2012.

[98] M. Hinze. A variational discretization concept in control constrained optimization: the linear-quadratic case. *Computational Optimization and Applications*, 30(1):45–61, 2005.

[99] M. Hinze, J. Krenciszek, and R. Pinnau. Proper orthogonal decomposition for free boundary value problems. *Hamburger Beiträge zur Angewandten Mathematik*, 2014.

[100] M. Hinze and K. Kunisch. Three control methods for time-dependent fluid flow. *Flow, Turbulence and Combustion*, 65(3-4):273–298, 2000.

[101] M. Hinze, R. Pinnau, M. Ulbrich, and S. Ulbrich. *Optimization with PDE constraints*, volume 23. Springer Science & Business Media, 2008.

[102] M. Hinze and S. Volkwein. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control. In *Dimension reduction of large-scale systems*, pages 261–306. Springer, 2005.

[103] M. Hinze and S. Volkwein. Error estimates for abstract linear–quadratic optimal control problems using proper orthogonal decomposition. *Computational Optimization and Applications*, 39(3):319–345, 2008.

[104] P. C. Hohenberg and B. I. Halperin. Theory of dynamic critical phenomena. *Reviews of Modern Physics*, 49(3):435–479, 1977.

[105] P. Holmes, J. L. Lumley, G. Berkooz, and C. W. Rowley. *Turbulence, coherent structures, dynamical systems and symmetry.* Cambridge University Press, 2012.

[106] R. H. Hoppe and Z. Liu. Snapshot location by error equilibration in proper orthogonal decomposition for linear and semilinear parabolic partial differential equations. *Journal of Numerical Mathematics*, 22(1):1–32, 2014.

[107] W. Huang and R. D. Russell. *Adaptive moving mesh methods*, volume 174. Springer Science & Business Media, 2010.

[108] A. Humphries and A. Stuart. Runge–Kutta methods for dissipative and gradient dynamical systems. *SIAM Journal on Numerical Analysis*, 31(5):1452–1485, 1994.

[109] T. Iliescu and Z. Wang. Are the snapshot difference quotients needed in the proper orthogonal decomposition? *SIAM Journal on Scientific Computing*, 36(3):A1221–A1250, 2014.

[110] A. Iollo, S. Lanteri, and J.-A. Désidéri. Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations. *Theoretical and Computational Fluid Dynamics*, 13(6):377–396, 2000.

[111] K. Ito and S. S. Ravindran. A reduced-order method for simulation and control of fluid flows. *Journal of Computational Physics*, 143(2):403–425, 1998.

[112] V. John. *Finite element methods for incompressible flow problems*, volume 51 of *Springer Series in Computational Mathematics.* Springer, Cham, 2016.

[113] V. John, A. Linke, C. Merdon, M. Neilan, and L. G. Rebholz. On the divergence constraint in mixed finite element methods for incompressible flows. *SIAM Review*, 59(3):492–544, 2017.

[114] C. Kahle. *Simulation and Control of Two-Phase Flow Using Diffuse Interface Models*. PhD thesis, Universität Hamburg, 2014.

[115] B. Karasözen and F. Yılmaz. Optimal boundary control of the unsteady Burgers equation with simultaneous space-time discretization. *Optimal Control Applications and Methods*, 35(4):423–434, 2014.

[116] M. Kärcher and M. Grepl. A certified reduced basis methods for parametrized elliptic optimal control problems. *ESAIM: Control, Optimisation and Calculus of Variations*, 20(2):416–441, 2014.

[117] A. Kolmogoroff. Über die beste Annäherung von Funktionen einer gegebenen Funktionenklasse. *Annals of Mathematics*, pages 107–110, 1936.

[118] M. E. Kowalski and J.-M. Jin. Model-order reduction of nonlinear models of electromagnetic phased-array hyperthermia. *IEEE Transactions on Biomedical Engineering*, 50(11):1243–1254, 2003.

[119] K. Kunisch and S. Volkwein. Control of the Burgers equation by a reduced-order approach using proper orthogonal decomposition. *Journal of optimization theory and applications*, 102(2):345–371, 1999.

[120] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic problems. *Numerische Mathematik*, 90(1):117–148, 2001.

[121] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM Journal on Numerical Analysis*, 40(2):492–515, 2002.

[122] K. Kunisch and S. Volkwein. Proper orthogonal decomposition for optimality systems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 42(1):1–23, 2008.

[123] K. Kunisch and S. Volkwein. Optimal snapshot location for computing POD basis functions. *ESAIM: Mathematical Modelling and Numerical Analysis*, 44(3):509–529, 2010.

[124] O. Ladyzhenskaya. *The mathematical theory of viscous incompressible flow*. Mathematics and its Applications, volume 2, Gordon and Breach, Science Publishers, 1969.

[125] J. Lang. *Adaptive multilevel solution of nonlinear parabolic PDE systems: theory, algorithm, and applications*, volume 16. Springer Science & Business Media, 2013.

[126] O. Lass. *Reduced order modeling and parameter identification for coupled nonlinear PDE systems*. PhD thesis, Universität Konstanz, 2014.

[127] O. Lass and S. Volkwein. Adaptive POD basis computation for parametrized nonlinear systems using optimal snapshot location. *Computational Optimization and Applications*, 58(3):645–677, 2014.

[128] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza. Model order reduction in fluid dynamics: challenges and perspectives. In *Reduced Order Methods for modeling and computational reduction*, pages 235–273. Springer, 2014.

[129] Y. T. Lee and A. A. Requicha. Algorithms for computing the volume and other integral properties of solids. I. Known methods and open issues. *Communications of the ACM*, 25(9):635–641, 1982.

[130] A. Linke. *Divergence-free mixed finite elements for the incompressible Navier-Stokes Equation*. PhD thesis, Friedrich-Alexander-Universität Erlangen-Nürnberg, 2007.

[131] J. L. Lions. *Optimal control of systems governed by partial differential equations*, volume 170. Springer Berlin, 1971.

[132] A. Logg, K.-A. Mardal, and G. Wells. *Automated solution of differential equations by the finite element method: The FEniCS book*, volume 84. Springer Science & Business Media, 2012.

[133] J. L. Lumley. The structure of inhomogeneous turbulence. *Atmospheric Turbulence and Wave Propagation*, pages 166–178, 1967.

[134] A. Massing, M. G. Larson, and A. Logg. Efficient implementation of finite element methods on nonmatching and overlapping meshes in three dimensions. *SIAM Journal on Scientific Computing*, 35(1):C23–C47, 2013.

[135] B. Mirtich. Fast and accurate computation of polyhedral mass properties. *Journal of graphics tools*, 1(2):31–50, 1996.

[136] W. F. Mitchell. Adaptive refinement for arbitrary finite element spaces with hierarchical bases. *Journal of Computational and Applied Mathematics*, 36(1):65–78, 1991.

[137] L. Modica. The gradient theory of phase transitions and the minimal interface criterion. *Archive for Rational Mechanics and Analysis*, 98(2):123–142, 1987.

[138] I. Neitzel, U. Prüfert, and T. Slawig. Solving time-dependent optimal control problems in COMSOL multiphysics. In *Proceedings of the COMSOL Conference Hannover*, 2008.

[139] I. Neitzel, U. Prüfert, and T. Slawig. A smooth regularization of the projection formula for constrained parabolic optimal control problems. *Numerical functional analysis and optimization*, 32(12):1283–1315, 2011.

[140] B. Nestler and A. Choudhury. Phase-field modeling of multi-component systems. *Current opinion in solid state and Materials Science*, 15(3):93–105, 2011.

[141] N. Nguyen, A. Patera, and J. Peraire. A ‘best points’ interpolation method for efficient approximation of parametrized functions. *International journal for numerical methods in engineering*, 73(4):521–543, 2008.

[142] P. Nigro, M. Anndif, Y. Teixeira, P. Pimenta, and P. Wriggers. An adaptive model order reduction by proper snapshot selection for nonlinear dynamical problems. *Computational Mechanics*, 57(4):537–554, 2016.

[143] Y. Oono and S. Puri. Study of phase-separation dynamics by use of cell dynamical systems. I. Modeling. *Physical Review A*, 38(1):434, 1988.

[144] G. M. Oxberry, T. Kostova-Vassilevska, W. Arrighi, and K. Chand. Limited-memory adaptive snapshot selection for proper orthogonal decomposition. *International Journal for Numerical Methods in Engineering*, 109(2):198–217, 2017.

[145] A. Pazy. *Semigroups of linear operators and applications to partial differential equations*, volume 44. Springer Science & Business Media, 1983.

[146] K. Pearson. LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11):559–572, 1901.

[147] A. D. Pelton. Thermodynamics and phase diagrams of materials. *Materials science and technology*, 2006.

[148] J. R. Phillips. Automated extraction of nonlinear circuit macromodels. In *Custom Integrated Circuits Conference, 2000. CICC. Proceedings of the IEEE 2000*, pages 451–454. IEEE, 2000.

[149] J. R. Phillips. Projection-based approaches for model reduction of weakly nonlinear, time-varying systems. *IEEE Transactions on computer-aided design of integrated circuits and systems*, 22(2):171–187, 2003.

[150] A. Pinkus. *N-widths in Approximation Theory*, volume 7. Springer Science & Business Media, 2012.

[151] R. Pinnau. Model reduction via proper orthogonal decomposition. In *Model Order Reduction: Theory, Research Aspects and Applications*, pages 95–109. Springer, 2008.

[152] R. W. Preisendorfer. Principal component analysis in meteorology and oceanography. *Elsevier Science Publishers*, 17:425, 1988.

[153] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced basis methods for partial differential equations: an introduction*, volume 92. Springer, 2015.

[154] J.-P. Raymond. Stokes and Navier–Stokes equations with nonhomogeneous boundary conditions. *Annales de l'Institut Henri Poincare – Non Linear Analysis*, 24(6):921 – 951, 2007.

[155] J. P. Raymond and H. Zidani. Hamiltonian Pontryagin's principles for control problems governed by semilinear parabolic equations. *Applied Mathematics and Optimization*, 39(2):143–177, 1999.

[156] M. Reed and B. Simon. *Methods of modern mathematical physics, I: Functional analysis.* Academic Press, 1980.

[157] M. Rewienski and J. White. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. *IEEE Transactions on computer-aided design of integrated circuits and systems*, 22(2):155–170, 2003.

[158] E. Rocca and J. Sprekels. Optimal distributed control of a nonlocal convective Cahn–Hilliard equation by the velocity in three dimensions. *SIAM Journal on Control and Optimization*, 53(3):1654–1680, 2015.

[159] A. Rosenfeld and A. C. Kak. *Digital picture processing.* Academic press, 1976.

[160] G. Rozza and K. Veroy. On the stability of the reduced basis method for Stokes equations in parametrized domains. *Computer methods in applied mechanics and engineering*, 196(7):1244–1260, 2007.

[161] A. Schmidt and K. G. Siebert. Design of Adaptive Finite Element Software: The Finite Element Toolbox ALBERTA. *Lecture Notes in Computational Science and Engineering*, (42), 2005.

[162] M. Schu. *Adaptive trust-region POD methods and their application in finance.* PhD thesis, University of Trier, 2012.

[163] A. J. Siade, M. Putti, and W. W.-G. Yeh. Snapshot selection for groundwater model reduction using proper orthogonal decomposition. *Water Resources Research*, 46(8), 2010.

[164] J. Simon. Compact sets in the space $L^p(0,t;B)$. *Annali di Matematica Pura ed Applicata (IV)*, 146(1):65–96, 1987.

[165] J. R. Singler. New POD error expressions, error bounds, and asymptotic results for reduced order models of parabolic PDEs. *SIAM Journal on Numerical Analysis*, 52(2):852–876, 2014.

[166] L. Sirovich. Turbulence and the dynamics of coherent structures. I. Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987.

[167] L. Sirovich. Turbulence and the dynamics of coherent structures. II. Symmetries and transformations. *Quarterly of Applied Mathematics*, 45(3):573–582, 1987.

[168] L. Sirovich. Turbulence and the dynamics of coherent structures. III. Dynamics and scaling. *Quarterly of Applied Mathematics*, 45(3):583–590, 1987.

[169] K. Steih. *Reduced basis methods for time-periodic parametric partial differential equations.* PhD thesis, Universität Ulm, 2014.

[170] I. Steinbach. Phase-field models in materials science. *Modelling and simulation in materials science and engineering*, 17(7):073001, 2009.

[171] R. Stevenson. Optimality of a standard adaptive finite element method. *Foundations of Computational Mathematics*, 7(2):245–269, 2007.

[172] J. E. Taylor and J. W. Cahn. Linking anisotropic sharp and diffuse surface motion laws via gradient flows. *Journal of Statistical Physics*, 77(1-2):183–197, 1994.

[173] R. Temam. *Navier-Stokes equations*, volume 2 of *Studies in Mathematics and its Applications.* North-Holland Publishing Co., Amsterdam-New York, 1979.

[174] J. B. Tenenbaum, V. De Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[175] Z. Tokoutsi, M. A. Grepl, K. Veroy-Grepl, M. Baragona, and R. Maessen. Real-Time Optimization of Thermal Ablation Treatments. In *Numerical Mathematics and Advanced Applications ENUMATH 2017*, pages 673–681. Springer International Publishing, 2019.

[176] F. Tröltzsch. *Optimal control of partial differential equations: theory, methods, and applications*, volume 112. American Mathematical Soc., 2010.

[177] F. Tröltzsch and S. Volkwein. POD a-posteriori error estimates for linear-quadratic optimal control problems. *Computational Optimization and Applications*, 44(1):83, 2009.

[178] P. G. Tucker. *Computation of unsteady internal flows: fundamental methods with case studies.* Springer Science & Business Media, 2012.

[179] S. Ulbrich and J. C. Ziems. Adaptive multilevel trust-region methods for time-dependent PDE-constrained optimization. *Portugaliae Mathematica*, 74(1):37–68, 2017.

[180] S. Ullmann. *POD-Galerkin Modeling for Incompressible Flows with Stochastic Boundary Conditions.* PhD thesis, Technische Universität Darmstadt, 2014.

[181] S. Ullmann, M. Rotkvic, and J. Lang. POD-Galerkin reduced-order modeling with adaptive finite element snapshots. *Journal of Computational Physics*, 325:244–258, 2016.

[182] M. Uzunca and B. Karasözen. Energy stable model order reduction for the Allen-Cahn equation. In *Model Reduction of Parametrized Systems*, pages 403–419. Springer, 2017.

[183] R. Verfürth. *A posteriori error estimation techniques for finite element methods.* Numerical mathematics and scientific computation. OUP Oxford, 2013.

[184] K. Veroy and A. Patera. Certified real-time solution of the parametrized steady incompressible Navier-Stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids*, 47(8-9):773–788, 2005.

[185] S. Volkwein. Optimal control of a phase-field model using proper orthogonal decomposition. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik: Applied Mathematics and Mechanics*, 81(2):83–97, 2001.

[186] S. Volkwein. Proper orthogonal decomposition: Theory and reduced-order modelling. *Lecture Notes, University of Konstanz*, 2013.

[187] S. Volkwein and S. Weiland. An algorithm for Galerkin projections in both time and spatial coordinates. *Proc. 17th MTNS*, 2006.

[188] Z. Wang. Nonlinear model reduction based on the finite element method with interpolated coefficients: semilinear parabolic equations. *Numerical Methods for Partial Differential Equations*, 31(6):1713–1741, 2015.

[189] A. Yagi. *Abstract parabolic evolution equations and their applications.* Springer Science & Business Media, 2009.

[190] M. Yano. A minimum-residual mixed reduced basis method: Exact residual certification and simultaneous finite-element reduced-basis refinement. *ESAIM: Mathematical Modelling and Numerical Analysis*, 50(1):163–185, 2016.

[191] M. Yano. *A reduced basis method with an exact solution certificate and spatio-parameter adaptivity: application to linear elasticity*, pages 55–76. Springer International Publishing, 2017.

[192] M. Yano. A reduced basis method for coercive equations with an exact solution certificate and spatio-parameter adaptivity: energy-norm and output error bounds. *SIAM Journal on Scientific Computing*, 40(1):A388–A420, 2018.

[193] F. N. Yilmaz. Space-time discretization of optimal control of Burgers equation using both discretize-then-optimize and optimize-then-discretize approaches. *PhD thesis, Middle East Technical University*, 2011.

[194] E. Zeidler. *Nonlinear functional analysis and its applications, I: fixed-point theorems.* Springer Verlag, 1986.

[195] E. Zeidler. *Nonlinear functional analysis and its applications, II/B: nonlinear monotone operators.* Springer Verlag, 1990.