On the POD Method

An Abstract Investigation with Applications to Reduced-Order Modeling and Suboptimal Control

Diplomarbeit

vorgelegt von

Markus Müller

aus

Kassel

angefertigt am

Institut für Numerische und Angewandte Mathematik

der

Georg-August-Universität zu Göttingen

2008

(überarbeitete Ausgabe)

To My Family

Preface

Dear Reader,

Throughout my studies, I have loved "mathematical truth", yet have never been pleased with the way it was presented. In fact, it has been appearing to me as an array of brilliant ideas buried under an inconvenient layout.
I therefore conclude

> Mathematical truth is not a matter of taste.
> Its presentation is.

This basically is the reason why I have put some effort into making my thesis "look" different. – And I felt I should better put some reflection on that matter in a preface.

Objectives of Layout Surely, I was concerned about a careful mathematical establishment of the theory. Yet, I also wanted to ensure that the reader is always aware of the *complete* picture of the matter. In particular, readers with little mathematical background should not be "left outside". In the best case, they should actually be motivated to "indulge" in the actual mathematics.
Furthermore, I assume that this thesis will be "referred to" rather than "being read". Therefore, the text should be structured in such a way that it provides a "key note list" such that "skimming" the thesis becomes easy.

The Notion of "True" and "Human" Maths I find that "true" mathematical derivation poses a problem to "human beings" due to different "directions of communication".
Let me illustrate this idea by "building a house" of bricks: From a mathematical point of view, it suffices to clearly define what to refer to as "a brick" and then denoting what to do with it – brick-wise. A problem now arises if the "idea" of a "house" *itself* is introduced in that manner (i.e. the recipient has got no understanding of what a "house" could be). At this point, there is some probability that after finishing the house, the reader will not be able to distinguish it from a "collection of bricks" that has been worked with.
For that reason, I have tried to always give an "idea of the house", i.e. the "complete picture" of the respective context. This of course may introduce some redundancy, yet I claim that the "length" of a mathematical script should not be measured in pages but in the time that it takes to understand it. The issue of providing a "complete picture" especially is helped by the following "method":

Properly Organizing Diagram – The "POD" Method In order to ensure orientation, I have added a notable number of diagrams that, to my knowledge, unfortunately do not have a tradition in mathematical writing. To ease navigation, some of these diagrams are linked to the text. In the case of the diagram of the "overall concept" of the thesis, we go along with Theodor Fontane:

> In case of correct structure, within the first page, the seed of it all ought to be contained.

In this thesis, this first "page" is given by the first figure, i.e. Figure 1. This is one way of overcoming the problem of "linear presentation" in a text. In that Figure, another approach to "non-linear presentation" becomes obvious: We establish the actual "cross paths" through the thesis, following the same "objective" respectively.

Boxes of "True" Maths In order to bring together "true" and "human" maths, all the "true" maths is put into boxes. Every box is labeled with its mathematical contents (say "Condition Number of..."). On the other hand, the corresponding paragraph heading gives an interpretation ("Sensitiveness of..." for example). In this way, it is also ensured that the structure of the document is determined by the interpretation rather than the "true" maths content (and consequently ensures the awareness of the "house"). Furthermore, the continuous use of paragraph headings shall simplify to skim through the issues of the thesis.

On the POD by "POD" The actual method of concern is the Proper Orthogonal Decomposition (POD). In our work group, the POD Method was not researched on before I started this thesis. Therefore, the role of my thesis initially was to investigate and discuss the method. Therefore, I technically had quite a bit of freedom in choosing the aspects of the method to care about. Out of interest, to better *understand* the method and to obtain a "critical" position, I chose not only to long for applications but also to investigate the abstract setting of the method.

Markus Müller, Göttingen, 03.03.2008

Acknowledgment

It is a pleasure to thank the many people who made this thesis possible – directly or indirectly. (All lists of names are given in alphabetical order of surnames.)

Dear Prof Lube,
Thank you very much indeed for changing my "mathematical" life from "I don't know what I really like in maths" to "I cannot write less pages on this – it's far too interesting – I'm sorry."
Thank you furthermore for actually realizing the "free spirit of university" – only giving me directions and not "telling me what to do" – yet still caring for my work very frequently and thoroughly.

Dear Prof Kunisch, dear Prof Volkwein,
Thank you very much for the kind invitation to Graz, being a fine host and in particular for carefully answering every single question I posed. I think this has improved this thesis a lot. Furthermore, I am very grateful for the opportunity of giving a talk on my thesis at the University of Graz.

Dear Student Colleagues (Marcisse Fuego, Timo Heister, Wiebke Lemster, Johannes Löwe, Lars Röhe and Benjamin Tews),
I am very grateful that you provided a stimulating and fun environment in which to learn and grow. In terms of this thesis, thank you for many concise enlightening discussions, proof-reading work and a hard to rival "spirit of effectiveness": "What about doing something useful?!"

Dear Dance Champions (Isabel Edvardsson, Franco Formica, Oxana Lebedew, Oksana Nikiforova, Evgen Vosnyuk, Marcus Weiss),
Many thanks for giving me inspiration and showing me an "alternative way of life".

Dear Friends (Victoria Fitz, Nilse Jäger, Zeus(i)),
Thank you for regular (philosophical) discussions on dance, life, education science or any other amusing endeavor (such as LaTeX). In terms of this thesis, many thanks for proof-reading the references.

Dear Family (Ma and Pa, Oma Olga and Oma Ursel, Schwester),
Thank you for your unconditional support: from providing healthy and tasty food boxes to careful proof-reading. – Not to mention the constant encouragement and love that I could rely on throughout my life. Furthermore, this thesis has only been possible since you have given me the opportunity to study. Thank you!

Abstract of Thesis

The objective of this thesis is an investigation of the *POD Method*. This method represents a parametrized set of data "better" than any other representation of its rank.
The POD Method is introduced and investigated in an abstract Hilbert space context. By optimally representing "snapshots" of a solution of an "Evolution Problem", a "POD Basis" is established. By means of this basis, a (Galerkin) *"reduced-order model"* is set up and used to reduce the (numerical) effort for "optimally controlling" the respective Evolution Problem. (This process is referred to as "*suboptimal control*".) Stress is put on a discussion of the chances and the effectiveness of the POD Method in terms of "Model Reduction".
On the "practical side", a specific example of the *non-stationary Heat Equation* is discussed in all the aspects mentioned and numerical implementations are done in Matlab and the Finite Element software Femlab.
In an appendix, aspects of the POD Method used in the Model Reduction process are enlightened by an exploration of the *statistical interpretation* of the method.

Contents

Chapter 0

Introduction

In this introduction, we proceed from the practical application to the "abstract mathematics" whereas in the thesis itself we start with the most abstract case and end with "practical" numerical applications. In this way, we ensure that we are *aware* of the "house" whilst carefully building it from scratch.

Orientation In this thesis, we pursue four objectives. Note that these objectives are depicted as gray boxes in Figure 1 and that they are as follows:

1. We wish to introduce the problem of Optimal Control (from a practical point of view) and consider basic solution theory.

2. The main objective shall be to investigate the concepts of Model Reduction and Suboptimal Control and in particular the role of the POD Method within them.

3. We wish to analyze and discuss the chances and obstacles of the POD Method in context of Model Reduction and Suboptimal Control.

4. On the theoretical side, we wish to *understand* the POD Method mathematically and phrase it in an abstract context.

In the remainder of this introduction, let us comment on the respective settings of these issues (the "What, Why and How"). Then, we shall link the topics to their actual location in the thesis (the "When") and finally get an overview of "new" results which could be established.

0.1 The What, Why and How

In this section, we shall present the issues of the thesis *"objective–wise"*. (In contrast to the following section, where we introduce the matters "topic-wise".) The number of the subsection refers to the number of the respective objective in the orientation. Each paragraph corresponds to one node in Figure 1 (apart from "Basic Idea of Way Out").

0.1.1 Optimal Control

Let us introduce the problem of Optimal Control of Evolution Problems and mention that in this thesis, we mathematically concentrate on linear-quadratic control problems for Evolution Problems with control constraints.

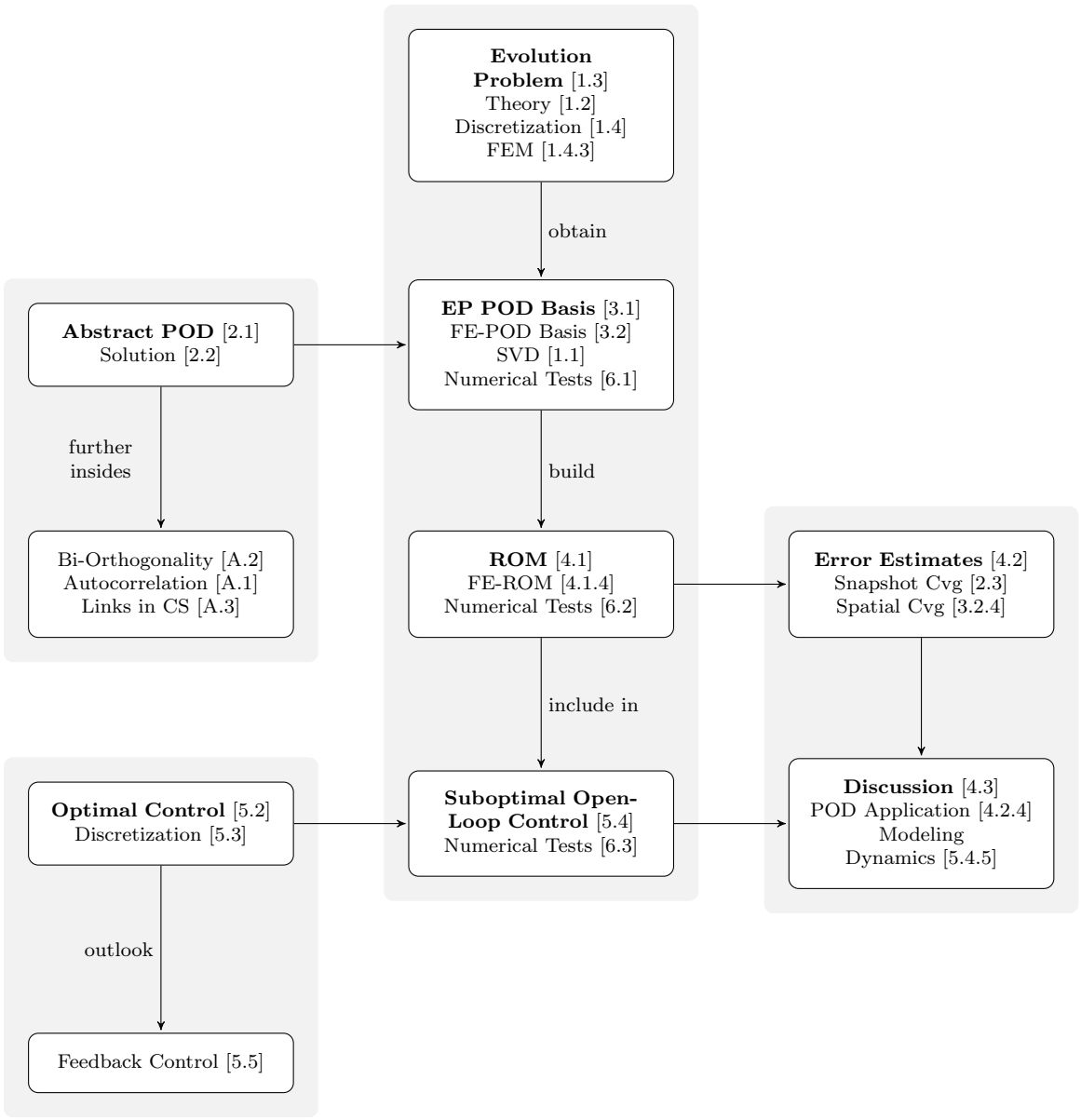

Figure 1: Structure of Thesis. All numbers refer to sections and subsections, respectively. Abbreviations: Coherent Structures (CS), Convergence (Cvg), Evolution Problem (EP), Finite Element Method (FEM), Proper Orthogonal Decomposition (POD), Reduced-Order Model (ROM).

Optimal Control In many cases, mathematical models of physical problems involve partial differential equations. These include problems such as the propagation of sound or heat, problems in fluid dynamics or the dynamics of electro-magnetic fields. In most cases, a solution may not be obtained analytically. Hence, enormous effort is put into approximating solutions numerically.
Having made good progress over the past decades in obtaining numerical solutions to these problems, the interest in *controlling* these equations has arisen. More formally speaking, one is interested in "controlling systems which are governed by these equations", i.e. in choosing data in the system such that the solution fulfills certain requirements. These requirements are represented in a so-called "cost functional" which is desired to be minimized.
In case of the Heat Equation which models the propagation of heat, we could ask for instance: What temperature should a heater have in order to yield a room temperature of approximately 20°C (without wasting energy, of course)?
Naturally, a fast solution of these problems is desired. Unfortunately, it turns out that the number of variables involved is typically *very* large – and hence, many problems are not feasible within a reasonable time frame. Thus, there is a huge demand to find possible "reductions" in the respective numerical effort – such as "Suboptimal Control" (see below).

Feedback Control As an outlook, we introduce the issue of feedback control since "immediate solutions" become even more important in this context: We wish to find a "rule" that – based on "measurements" of the state of a system – determines an optimal choice of a control variable. In particular, we focus on the so-called linear-quadratic regulator problem.

0.1.2 Suboptimal Control of Evolution Problems

Having learned that Optimal Control problems are important to many applications but often hard to tackle numerically, let us concentrate on reducing the effort of such calculations. In particular, we wish to setup a reduced-order model and Suboptimal Control strategies for a certain class of problems.
In this subsection, we wish to explain the idea of the procedure and outline the role of POD within it. Furthermore, we aim to specify the class of problems of concern.

Basic Idea of Way Out In order to calculate a solution of an optimal control problem, it needs to be discretized. Usually, the corresponding choice of spatial basis functions is "general", i.e. independent of the actual problem of concern. Thus, a high number of degrees of freedom (a high "rank of approximation") has to be used in order to obtain satisfying approximations.
For that reason, the numerical effort to compute an optimal control may be reduced by introducing "intelligent" basis functions which have got some "knowledge" about the expected solution. A solution to this reduced problem is then called "suboptimal control" since it only is optimal within the "knowledge base" of these basis functions.

(Discretized) Evolution Problem In this thesis, we consider an Evolution Problem of the form:

$$\begin{aligned} \frac{d}{dt}\left(y(t),\varphi\right)_H + a(y(t),\varphi) &= \left(F(t),\varphi\right)_H, \\ y(0) &= y_0 \quad \text{in } H, \end{aligned}$$

where a denotes a *symmetric* bilinear form and F as well as the solution y are "abstract functions". The equation should hold for all $\varphi \in V$, where V denotes a so-called "ansatz space". As a special case of this problem, we consider a parabolic initial value problem: For suitable coefficient collections a and c as well as a right-hand side f, we wish to find a (sufficiently smooth) function y on the

time-space domain $Q_T := (0,T) \times \Omega \subset \mathbb{R} \times \mathbb{R}^n$, $n = 2, 3$, such that there holds

$$\begin{aligned} \frac{\partial y}{\partial t} - \sum_{i,j=1}^{n} \frac{\partial}{\partial x_i} \left(a_{ij} \frac{\partial y}{\partial x_j} \right) + cy &= f && \text{in } Q_T, \\ y(t,x) &= 0 && \text{on } (0,T) \times \partial\Omega, \\ y(0,x) &= y_0(x) && \text{in } \{0\} \times \Omega. \end{aligned}$$

In order to calculate solutions to these problems, we find discrete formulations in a so-called "vertical way", i.e. we carry out a (Galerkin type) space-discretization first and then discretize in time. (This order of procedure shall turn out to be of some importance for the construction of the POD Method.) In particular, we use a Finite Element discretization in space and an implicit Euler method in time.

Obtaining an "Intelligent" Basis – The Role of POD As mentioned above, we seek for an "intelligent" spatial basis in order to reduce the "rank of approximation". Such an "intelligent" basis may be obtained by "optimally" representing *snapshots* of the "state" of the system, i.e. by taking solutions at certain time instances and then finding their *key ingredients*. In this sense, the basis has got "knowledge" about characteristics of the system and hence there is a *hope* that it may represent the dynamics with fewer basis elements than a "general" basis would be able to.
Establishing such an *optimal representation of a snapshot set* is exactly the aim of the POD Method (in context of Model Reduction). In particular, the POD Method finds *orthogonal* basis elements, which optimally represent the snapshot set (in the quadratic mean), i.e., it establishes a *"Proper Orthogonal Decomposition"* which actually has given the method its name. The resulting POD Basis elements in this sense are "tailored" to a particular solution of the Evolution System.
We deduce the calculation of such modes from the general theory on the POD Method. In numerical examples, we consider a well-suited snapshot set (made up of Fourier modes) as well as a more challenging example. In particular, we wish to get an understanding of what the POD Method is capable of achieving in terms of "representation of snapshots". Furthermore, we study whether subtracting the mean from the snapshots improves the results.

Reduced-order Modeling Having obtained a POD Basis, we may setup a respective Galerkin model for the Evolution Problem of concern. Since the dimension of the POD Basis is smaller, we have to determine *fewer* "coefficients" and hence we call this model a "reduced-order" model.
We find three ways of benefiting from POD-ROM in practice: extra- or interpolation of a given solution or computing a solution by means of a known solution (to a system with slightly different data). We illustrate these ways by means of numerical calculations for the *non-stationary Heat Equation* (which is a special case of the parabolic initial value problem introduced above).
Furthermore, we carry out a numerical study concerning the influence of the discretization in the snapshots on the resulting POD Basis.

Suboptimal Control We consider an Optimal Control problem (see above) which involves the Evolution Problem of concern and reduce the effort of calculating a solution by discretizing the problem with a "POD Galerkin scheme", i.e. we apply the reduced-order model constructed.
Unfortunately, we find a difficulty in this context: We are to calculate a control whose corresponding "state" (solution of the Evolution Problem) shall be optimal but is of course not known a priori. On the other hand, a POD Basis is tailored to a particular solution, which therefore has to be provided at some point. Yet in order to calculate a solution, we have to set ("guess") an actual value for the control variable. A priori, there is no guarantee that the resulting state bears the same characteristics as the "optimal" state would do. Hence, we cannot tell whether the POD Basis chosen would be able to model the "optimal state" at all. We shall work out ways to overcome this problem of "non-modeled dynamics".
Finally, we illustrate the suboptimal control strategy by means of (simplified) numerical examples. We consider *distributed* control of the non-stationary Heat Equation for "tracking" a *final-time target*

as well as a target state for the *whole* time interval. In particular we (again) choose a "Fourier example" as well as a more challenging case.

0.1.3 (Mathematical) Discussion – Problems Posed

Having setup a suboptimal control strategy, we of course wish to investigate it *mathematically*.

Asymptotic Behaviour – Error Estimates In fact, there is a variety of sources of errors in the procedure of reduced-order modeling. In particular we cannot obtain error estimates by approximation results in suitable function spaces.
Therefore, we shall determine estimates for a basic case and take care of further sources by an "asymptotic analysis": We wish to find out whether the locations of snapshots matter "in the limit" and whether we may control (numerical) errors in the snapshots themselves.

Discussion Based on a mathematical investigation, we shall try to find answers to the following questions

- In which regard is POD optimal?
- What are benefits and drawbacks of POD as a Model Reduction tool?
- How to choose a snapshot grid and how to obtain snapshots? That implies: How to locate instances in time of characteristic dynamics? How to establish a solution at these time instances?
- How to predict the quality of a reduced-order solution?
- How to tackle the problem of non-modeled dynamics in suboptimal control?

0.1.4 Understanding the POD Method

Regularly, people refer to the *whole process* of Model Reduction as "applying POD". Yet the POD Method actually presents only a small portion of the process of Model Reduction: it only may represent a certain snapshot set "optimally".
As far as this application is concerned, the theory of the POD Method may be discussed in one sentence: We wish to optimally represent snapshots in the quadratic mean, yet this is a well-known property of so-called singular vectors of a suitable matrix.
On the other hand, this level of insight does not suffice to actually answer any of the questions posed above, for instance. Thus, we also wish to focus on the POD Method in a somewhat more *abstract* sense – in order to explore "connections" which would remain hidden otherwise (on the level of matrices for example). Basically, we pursue three objectives:

1. We wish to gain an actual *understanding* of the POD Method which shall also help us to answer some of the question posed above.
2. We investigate the POD Method in an abstract context, independent of the setting of Evolution Problems. This shall then enable us to deduce problem statements and solutions for the various contexts in which the POD Method shall be appearing as we proceed.
3. We wish to point out links to other "focuses" of application of the POD Method of which we may benefit at some point.

POD for "Abstract Ensembles" In an "abstract" setting, we present the POD Method for finding a representation of *parametrized data* in a "general" Hilbert space. We motivate the actual ingredients of the "POD Problem" and characterize its solution mathematically. All these investigations serve as a basis to derive the concrete cases from.

Statistical Concepts of the POD Method We focus on the statistical background of the POD and make the setting more concrete by choosing the Hilbert space to be a function space. In this way, we introduce a second parameter of the data and may then find that the POD modes are part of a "bi-orthogonal decomposition". (This shall aid understanding the so-called "Method of Snapshots" for instance.)
We shall then rephrase the characterizing operator of a POD Basis as an "autocorrelation" operator. This shall yield hints in which situations the POD Method shall struggle to provide pleasing results. Furthermore, we may *interpret* the POD modes to decompose the "autocorrelation operator" and conclude that POD may be used as a tool to detect "Coherent Structures" in (say) a fluid flow.
Links to other approaches (such as the Fourier decomposition) and the role of the "statistical" POD in numerical calculations shall complete the picture.

0.2 The When

Having introduced the issues of the thesis "objective-wise", let us now link these objectives to their actual place of treatment in the thesis.

Chapter 1: Basics In this "introductory" chapter, we introduce the mathematical ingredients in order to formulate the Evolution Problem of concern and comment on its discretization. We explain that parabolic Initial Value Problems lead to such Evolution Problems by means of a *"variational formulation"*. We show their discretization in space by means of the Finite Element method in space and by the backward Euler method in time.

Chapter 2: Abstract POD We transform the idea of the POD Method into mathematical language and investigate the method for ensembles lying in a "general" Hilbert space and characterize its solution in two ways. Furthermore, we ensure the existence of a POD Basis and derive an error estimate of the POD approximation of the corresponding ensemble.

Chapter 3: POD for ROM We apply the abstract theory on the POD Method to the context of Evolution Problems, the focus being the application in Reduced-Order Modeling. In particular, we choose the POD ensemble to be a *"snapshots set"*. Furthermore, we show how a POD Basis may be obtained on a Finite Element level.

Chapter 4: Reduced-order Models We introduce reduced-order models as a special sort of Galerkin discretization. We then carry out a thorough error analysis leading to two types of estimates. We also simplify and improve these estimates. We conclude with a discussion on the POD Method as a Model Reduction tool.

Chapter 5: (Sub) Optimal Control We introduce the concept of Optimal Control for Evolution Problems and comment on respective numerical strategies. We make then use of the reduced-order models developed in Chapter 4 in order to reduce the numerical costs of the Optimal Control problem. We point out the problems which potentially appear in this approach.

Chapter 6: Numerical Experiments Our primary objective of carrying out numerical experiments shall be to *illustrate* the theory developed. For that purpose, we shall choose examples which are simple enough to be understood also by people who have not worked on Model Reduction so far. Essentially, a "feeling" for what "POD is capable of doing and what it is not" shall be communicated. We also further "investigate" the method at a practical level. In particular, we explore the consequences of mean subtraction in the snapshot set before obtaining a POD Basis. Furthermore, we present a basic study on the dependence of a POD Basis on the discretization of the system that yields the snapshot set.

Chapter 7: Summary, Discussion and Outlook We summarize the findings along the lines of Figure 1. We discuss the POD Method as a Model Reduction tool as well as a strategy in Suboptimal Control. Finally, we give an outlook on what "could have been done" and on what "should be done" (based on the findings of the discussion).

Appendix A: Statistical POD We choose the Hilbert space of Chapter 2 to be a function space over a domain Ω. Thus, the level of abstraction is between those of Chapters 2 and 3. We show that POD modes are part of a "bi-orthogonal decomposition" of an ensemble. We interpret the POD operator as a "correlation operator" and point out links to other decomposition schemes. We summarize the role of the statistical background of the POD in the process of the numerical treatment of Evolution Problems. Finally, we show that hence the POD Method may be used to establish so-called "Coherent Structures" (in actually two different senses). (Since most parts of this chapter are "off the main track" of the thesis, we have placed it in appendix.)

0.3 The What's New

It shall not be concealed that most parts of this thesis present a "survey" of research literature. Anyhow, along the lines of this survey, quite a few "improvements" and "new" points of view could be found. Since in the full summary of Chapter 7 these facts become less obvious, let us gather these findings at this point:

0.3.1 General Improvements

Proofs Elaborated Since research literature tends to be very concise, a "global" achievement surely is to present proofs and explanations in a fashion suitable for "pre-researches" – especially in terms of the error analysis of POD reduced-order models.

Illustration and Layout Apart from carefully elaborating the proofs, the "understanding" of the issues shall be helped by diverse diagrams which all have not been found in the literature. Indirectly, on a general level, a layout of "mathematical writing" is proposed that *visually* puts the "mathematics" in the center of argumentation, but, at the same time, smoothly blends into the overall structure of the document.

0.3.2 POD

Clarification of the POD Problem Statement A POD Problem is introduced in a way that enables the reader to properly *understand* its ingredients. The stress lies on the *interpretation* of "mathematical symbols" (such as sum operations, which may be due to (say) an averaging operation or (say) a linear combination). Certain components are discussed in further detail (the "optimality norm" and the average operator, for instance). Furthermore, a link of the vertical approach of discretization to the way of application of the POD Method is drawn ("key spatial" vs "key temporal" structures).

POD Method in Abstract Setting The POD Method is formulated in an abstract setting in order to "decouple" it from the application to Evolution Problems. In particular, the notion of an average operator is introduced and "formalized". In the short term view, this helps to deduce all concrete occurrences of the method from this abstract case.
In a longterm view, this may aid merging the "numerical" and the "statistical" interpretation of the POD Method (refer below).

Merging POD Approaches: Numerical Calculation vs Statistical Understanding The POD Method generally is applied in two different contexts: the investigation of the "overall behaviour" of "dynamical systems" and reduced-order modeling of "variational formulations" of (say) Evolution Problems. Basically, in the first approach, people are interested in *"statistically" understanding* the system at hand. In the latter approach, people wish to *use* the POD Basis (in a reduced-order model say) in order to *"numerically" calculate* a solution of the system.
An explanation of a typical aspect of the latter area ("Method of Snapshots") could be given by a typical aspect of the former area ("bi-orthogonal decomposition of signals"). Vice versa, the equivalence of two definitions of Coherent Structures is shown by virtue of a proof in Volkwein 1999 (which deals with the POD Method from a "variational" point of view). Furthermore, the roles of the statistical concepts of the POD in a numerical application of the method are clearly outlined.

0.3.3 Reduced-Order Modeling

Alternative Error Estimate In the error analysis of POD reduced-order models, a technique of estimating the "z-term" in Volkwein 2006 was applied to the more general case of Kunisch and Volkwein 2002.

Investigation of Practical Scenarios of Applying POD-ROM Practical scenarios of applying the POD Method in context of Model Reduction are proposed and tested for (basic) numerical examples: interpolating a given solution in time, extrapolating a given solution in time and setting up a reduced-order model based on "perturbed snapshots", i.e. on snapshots that are obtained from a system that is different to the system to be solved.
The requirements on a corresponding error analysis are outlined – the primary objective being to find error estimates that let the user *predict* the quality of a low-order solution a priori. A connection to the (known) "ideal" error estimate is drawn and corresponding future work outlined.

Chapter 1

Mathematical Basics

In this introductory chapter, we wish to present the mathematical background of the issues carried out in the remainder of the thesis. In particular, we introduce the Singular Value Decomposition (SVD) as well as some results of Functional Analysis and work on the theory for the Evolution Problems of consideration in Reduced-Order Modeling and optimization. Throughout, we only present those parts of the theory actually needed in the chapters following. Way more details may of course be found in the respective references.

Prerequisites Throughout the thesis, we denote the transpose of a matrix A by A^T. Furthermore, we assume the following basic issues of linear algebra to be familiar: the notion of an orthogonal matrix, the root of a matrix and a "convex hull", the Schwarz' and Young's inequality as well as the Proposition of Riesz.

1.1 Singular Value Decomposition of Matrices

It shall turn out that in a "discrete context" a POD Basis may be found by means of a Singular Value Decomposition (SVD) of a so-called *Ensemble Matrix*. Hence, we wish to give some background on the method.

1.1.1 Theory on the SVD of Matrices

Let us first quote some theoretical results on the issue of SVD.

Ingredients The following terms are essential to an SVD.

Definition 1.1.1 (Singular Values and Singular Vectors)
For $m \geq n$, let $Y \in \mathbb{R}^{m\times n}$ be a matrix of rank $d \leq n$. Let $U := \{u_i\}_{i=1}^m \subset \mathbb{R}^m$ and $V := \{v_i\}_{i=1}^n \subset \mathbb{R}^n$ be sets of pair-wise *orthonormal* vectors such that

$$Yv_i = \sigma_i u_i \quad \text{and} \quad Y^T u_i = \sigma_i v_i \quad \text{for } i = 1, \ldots, d. \tag{1.1}$$

Then, $\sigma_1, \ldots, \sigma_d$ are called *singular values*, all $u \in U$ are called *right singular vectors* and all $v \in V$ are called *left singular vectors*.

Existence and Uniqueness We show the existence of such an SVD and in the same breath restate it in a "matrix fashion". Furthermore, we discuss whether the decomposition is uniquely determined.

Proposition 1.1.2 (Existence of an SVD)
Let $Y = [y_1, \ldots, y_n]$ be a real-valued $m \times n$ matrix of rank $d \leq \min\{m, n\}$. Then, there exists a *Singular Value Decomposition* of Y, i.e. real numbers $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_d > 0$ and orthogonal matrices $U = [u_1, \ldots, u_m] \in \mathbb{R}^{m \times m}$ and $V = [v_1, \ldots, v_n] \in \mathbb{R}^{n \times n}$ such that:

$$Y = U \Sigma V^T, \quad \Sigma := \begin{pmatrix} D & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{m \times n},$$

where $D = diag(\sigma_1, \ldots, \sigma_d) \in \mathbb{R}^{d \times d}$ and the 0-blocks are of appropriate dimensions.

Proof.
A proof of this proposition might be found in Stewart 1973, Theorem 6.1 or Antoulas 2005, Theorem 3.3. In order to prove that this restatement actually yields an SVD in the sense of Definition 1.1.1, we show that (1.1) holds: Since U and V are orthogonal matrices, the claim simply follows from $YV = U\Sigma V^T V = U\Sigma$ and $Y^T U = V \Sigma U^T U = \Sigma V$. □

Lemma 1.1.3 (Discussion of Uniqueness)
For an arbitrary matrix $Y \in R^{m \times n}$, the singular values are unique. Only the left and right singular vectors corresponding to non-zero singular values of multiplicity one are unique – and only determined up to simultaneous sign changes.

Proof.
Refer to Stewart 1973, p. 319. □

Essential Property Later on, we shall use the fact that finding an SVD may be transformed into an eigenvalue problem.

Remark 1.1.4 (Singular Vectors as Eigenvectors)
By inserting the equations of (1.1) into each other, we find: The right singular vectors $\{u_i\}_{i=1}^d$ are *eigenvectors of* YY^T to the eigenvalues $\lambda_i = \sigma_i^2$ and the left singular vectors $\{v_i\}_{i=1}^d$ are *eigenvectors of* $Y^T Y$ to the eigenvalues $\lambda_i = \sigma_i^2$, i.e.,

$$YY^T u_i = \sigma_i^2 u_i \quad \text{and} \quad Y^T Y v_i = \sigma_i^2 v_i \quad \text{for } i = 1, \ldots, d$$

and for $i > d$ we obtain $YY^T u_i = Y^T Y v_i = 0$.

Geometric Interpretation In order to enlighten its nature, let us look at the SVD from a geometric point of view: From (1.1) we may derive the following "visualization" of the SVD of an $m \times n$ matrix $Y = U\Sigma V^T$: Consider the image of the unit sphere under Y, i.e. an ellipsoid. Then, we may identify the *columns of* V with the *principal axes* of the ellipsoid, the *columns of* U with the *preimages of the principal axes* and the *singular values* with the *principal radii*. For $n = m = 2$, these relations are depicted in Figure 1.1; alternatively, see Antoulas 2005, Figure 3.2.

Relationship of Eigenvalue Decomposition and SVD In some sense, the SVD may be seen as a generalized eigenvalue decomposition as it may be applied to an arbitrary matrix A (not necessarily square) and still always leads to real singular values and the singular vectors are orthogonal. On the other hand, these vectors are not even "dimension invariant" under A, whereas its eigenvectors would be "direction invariant". For more details on the comparison of SVD and eingenvalue decomposition, refer to Antoulas 2005, Subsection 3.2.3. We only quote a result on the relationship of eigen- and singular values in the special case of A being symmetric:

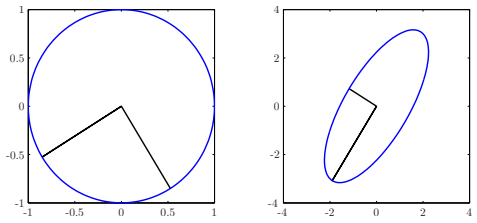

Figure 1.1: Visualization of the geometric interpretation of singular vectors and singular values of a 2×2 matrix. Unit circle C and right singular vectors v_i (left). Image of C under Y and vectors $\sigma_i u_i$ (right).

Lemma 1.1.5 (Eigenvalues vs Singular Values)
Let $A \in \mathbb{R}^{n\times n}$ a *hermitian* matrix with eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_n$. Then, the singular values of A are $|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|$.

Proof.
Refer to Stewart 1973, Theorem 6.3. □

Sensitivity to Coordinate Changes The following fact shall be of some importance in the discussion of the POD Method (refer to Section 4.3): Changes to the coordinates in finite dimensional vector spaces can be realized by a "change of basis" matrix. So the following lemma imposes that in such situations some care has to be taken.

Lemma 1.1.6 (SVD not invariant)
Let A be a $m \times n$ matrix and B a regular $n \times n$ matrix. Then, in general, A and AB do not have the same singular values necessarily. But these are uniquely determined. In this sense, we may say

$$\text{“SVD}(A)\text{”} \neq \text{“SVD}(AB)\text{”}.$$

Proof.
As a counter example, we choose $B \in \mathbb{R}^{n\times n}$ to be two times the identity in $\mathbb{R}^{n\times n}$. Clearly, B is regular, but the singular values of AB are two times the singular values of A. □

1.1.2 Optimal Approximation Property

We conclude this section by stating the *essential property* of SVD in terms of application to the POD Method. This result is also known as *Schmidt-Eckart-Young-Mirsky-Theorem*.

Theorem 1.1.7 (Optimal Approximation)
For $m \geq n$, let $A \in \mathbb{R}^{m\times n}$ be a matrix of rank $d \leq n$. Let $A = U\Sigma V^T$ be an SVD and $\sigma_1, \sigma_2, \ldots, \sigma_n$ the singular values of A. Construct an *"approximation of rank ℓ"* A^ℓ by setting $\sigma_{\ell+1} = \sigma_{\ell+2} = \cdots = \sigma_n = 0$.
Then, we find that A^ℓ in the Frobenius-Norm is the *unique* best approximation to A amongst all matrices of rank ℓ:

$$\left\|A - A^\ell\right\|_{\text{Fro}} = \min_{\text{rank}(B)=\ell} \|A - B\|_{\text{Fro}} = \left(\sum_{i=\ell+1}^{d} \sigma_i^2\right)^{\frac{1}{2}}.$$

The same statement holds true for the $\|\cdot\|_2$-Norm with the minimal value $\sigma_{\ell+1}$, but in this case A^ℓ is not the *unique* minimizer anymore.

Proof.
Refer to Stewart 1973, Theorem 6.7 or to Antoulas 2005, Theorem 3.6, Remark 3.2.2. □

1.2 Functional Analysis

We provide some basic results from functional analysis which we shall need for the establishment of a solution of the so-called Evolution Problems of concern.

1.2.1 Basic Definitions

We assume the following terms to be familiar: vector space, norm, inner product, Banach space, Hilbert space, Lebesgue measurable set. Furthermore, we do not explicitly define the notion of a "dual space" H^* of a Hilbert space H as well as the corresponding "duality pairing" $\langle \cdot, \cdot \rangle_{H^*,H}$.

Hilbert Space of Essentially Countable Dimension In some cases, we shall have to restrict problems to Hilbert spaces of "nearly countable" dimension in the following sense:

Definition 1.2.1 (Separable Hilbert Space)
A Hilbert Space V is separable if there is a basis $\{\varphi_j\}_{j\in\mathbb{N}}$ in V and for all $v \in V$ there exists a sequence $\{v_n\}_{n\in\mathbb{N}} \subset \operatorname{span}(\varphi_1, \ldots, \varphi_n)$ with

$$\lim_{n\to\infty} \|v - v_n\|_V = 0.$$

Orthogonal Projection Orthogonal projections play a crucial role in context of the POD Method. Hence, let us define them and give an alternative representation (a corresponding proof might be found in Lube 2005).

Definition 1.2.2 (Orthogonal Projection and its Fourier Representation)
Let $(X, (\cdot,\cdot)_X)$ be a Hilbert space with norm $\|\cdot\|_X$ induced by the inner product. Let X^n be a separable, closed subspace of X. We call an operator P an *orthogonal projection* on X^n, if

$$P : X \to X^n, \quad (P\varphi, \psi)_X = (\varphi, \psi)_X \quad \text{for all } \psi \in X^n.$$

In case that $\dim X^n = n < \infty$ and $\{\psi_i\}_{i=1}^n$ is an *orthonormal* basis of X^n, we may write the projection P in *Fourier representation* form, given by

$$F : X \to X^n, \quad F(\varphi) = \sum_{i\in\mathbb{N}} (\varphi, \psi_i)_X \, \psi_i \quad \text{for all } \varphi \in X.$$

1.2.2 Theory on Partial Differential Equations

Essentially, we shall introduce all the spaces of functions necessary in order to establish a theory of solution of the Evolution Problems of concern.

Spaces for Integration Let us first define the spaces of functions which are integrable in a certain sense and characterize them as Banach spaces.

Definition 1.2.3 (Lebesgue Spaces)
Let $\Omega \subset \mathbb{R}^n$ be a bounded, Lebesgue measurable set and let $|\Omega|$ denote the n-dimensional *Lebesgue-measure*. Then, define for $1 \leq p < \infty$

$$L^p(\Omega) := \Big\{ u : \Omega \to \mathbb{R} \quad \text{with} \quad \|u\|^p_{L^p(\Omega)} := \int_\Omega |u(x)|^p \, \mathrm{d}x < \infty \Big\}.$$

and for $p = \infty$

$$L^\infty(\Omega) := \Big\{ u : \Omega \to \mathbb{R} \quad \text{with} \quad \|u\|^p_{L^\infty(\Omega)} := \operatorname*{ess\,sup}_{x \in \Omega} |u(x)| < \infty \Big\}.$$

For all $1 \leq p \leq \infty$ define a "localization"

$$L^p_{\mathrm{loc}}(\Omega) := \{ u : \Omega \to \mathbb{R} \quad \text{with} \quad u \in L^p(\Omega_0) \text{ for any open } \Omega_0 \subset\subset \Omega \},$$

where $\Omega_0 \subset\subset \Omega$ denotes a *compact* subset Ω_0 of Ω.

Lemma 1.2.4 (L^p is Banach, L^2 is separable Hilbert Space)
The spaces $L^p(\Omega), 1 \leq p \leq \infty$ endowed with the respective norms are Banach spaces. $L^2(\Omega)$ endowed with the inner product $(u, v)_{L^2(\Omega)} := \int_\Omega u(x)v(x) \, \mathrm{d}x$ is a separable Hilbert space.

Proof.
Refer to Alt 1992, Satz 1.17, Lemma 1.13 and Bemerkung 1.12. Furthermore, any space $L^p(\Omega), 1 \leq p < \infty$ is separable according to Dobrowolski 2006, Satz 4.20(b). □

Spaces for Differentiation We shall simplify the notation of "all derivatives up to a certain order" and then state the space of all (k-times) differentiable functions.

Definition 1.2.5 (Derivatives with Multiindeces)
For a *multiindex* $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$, its order $|\alpha| := \alpha_1 + \cdots + \alpha_n$ and a sufficiently differentiable function u define

$$D^\alpha u := \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \cdots \partial x_n^{\alpha_n}} \, u,$$

where $\partial x_i^{\alpha_i}$ denotes $\partial x_i \cdots \partial x_i$ (α_i times). We also define $D^0 u := u$.

Definition 1.2.6 (Spaces $C^k(\Omega)$)
Let $\Omega \subset \mathbb{R}^n$ be a domain. For $m \in \mathbb{N}_0$ define $C^m(\Omega)$ of all m-times differentiable functions $u : \Omega \to \mathbb{R}$ and let $C^\infty(\Omega) := \bigcap_{m=0}^\infty C^m(\Omega)$ be the space of all infinitely differentiable functions.
Let $C_0^\infty(\Omega)$ denote the subspaces of $C^\infty(\Omega)$, consisting only of those functions with compact support in Ω, i.e.

$$C_0^\infty(\Omega) := \{ f \in C^\infty(\Omega) \mid \overline{\{x \in \Omega \mid f(x) \neq 0\}} \subset \Omega \text{ is compact} \}.$$

Weak Differentiation As for most applications we wish to weaken the notion of a derivative, we define a derivative which is only determined in terms of an "integral equation". In case of existence, this *"weak derivative"* is uniquely determined due to Dobrowolski 2006, Lemma 5.4 for example.

By means of this concept, we may then introduce spaces of functions which are weakly differentiable and characterize them as Hilbert spaces. These spaces shall turn out to be fundamental to the theory of the solution of Evolution Problems. In particular, we introduce a space $H_0^1(\Omega)$, which is suitable for our later choice of boundary values to be zero (in a weak sense).

Definition 1.2.7 (Weak Derivative)
Let $u \in L^1_{\text{loc}}(\Omega)$. If there exists a function $u_\alpha \in L^1_{\text{loc}}(\Omega)$ such that

$$\int_\Omega D^\alpha \varphi \, \mathrm{d}x = (-1)^{|\alpha|} \int_\Omega u_\alpha \varphi \, \mathrm{d}x \quad \text{for all } \varphi \in C_0^\infty(\Omega),$$

we call u_α the *α-th weak derivative* of u in Ω.

Definition 1.2.8 (Sobolev Spaces $H^k(\Omega)$ and $H_0^1(\Omega)$)
Let $\Omega \in R^n$ be a bounded domain. Let D^α denote the multiindexed derivative in the weak sense of Definition 1.2.7. Define the *Sobolev Space*

$$H^k(\Omega) = \{v \in L^2(\Omega) \mid \text{there exists } D^\alpha v \in L^2(\Omega) for\, |\alpha| \leq k\}$$

and endow it with the inner product (inducing a norm $\|\cdot\|_{H^k(\Omega)}$)

$$(v, w)_{H^k(\Omega)} = \int_\Omega \sum_{|\alpha| \leq k} \partial^\alpha v \; \partial^\alpha v \; \mathrm{d}x.$$

Note that in particular for $k = 1$ and $|v|_1^2 := \int_\Omega |\nabla v|^2 \, \mathrm{d}x$ we obtain:

$$H^1(\Omega) = \{v \in L^2(\Omega) \mid \text{there exists } \nabla v \in [L^2(\Omega)]^3\} \quad \text{with} \quad \|v\|^2_{H^1(\Omega)} = \|v\|^2_{L^2(\Omega)} + |v|_1^2 .$$

Furthermore, let $H_0^1(\Omega)$ be the completion of $C_0^\infty(\Omega) \cap H^1(\Omega)$ with respect to the norm $\|\cdot\|_{H^k(\Omega)}$ which by definition is a Banach space.

Lemma 1.2.9 ($(H^k(\Omega), (\cdot,\cdot)_{H^k(\Omega)})$ is a Hilbert Space)
For all $k \in \mathbb{N}$, $H^k(\Omega)$ endowed with the norm $\|\cdot\|_{H^k(\Omega)}$ is a Hilbert space.

Proof.
Refer to Knabner and Angermann 2000, Satz 3.3. □

Solution Theory Central to the solution theory of the problems of consideration later on, shall be the Lemma of Lax-Milgram, for which we introduce the notion of a coercive operator.

Definition 1.2.10 (Coercive Operator)
Let X be a Hilbert space. An operator $A \in L(X, X)$ is called *strictly coercive* on X (or X-elliptic) if there exists a constant $\gamma > 0$ such that there holds

$$Re(Av, v) \geq \gamma \|v\|_X^2 \quad \text{for all } v \in X.$$

Lemma 1.2.11 (Lemma of Lax-Milgram)
Let X be a Hilbert space and let $A \in L(X, X)$ be a strictly coercive operator.
Then, there exists the inverse operator $A^{-1} \in L(X, X)$.

1.2.3 Theory for Evolution Problems

We shall introduce appropriate spaces for the solution of Evolution Problems. The spaces introduced in the theory for PDE only contain functions which are defined in the (spatial) domain Ω. Naturally, the solution of an Evolution Problem depends on time. For that reason, we shall define spaces of so-called *abstract functions*. (Note that this concept reflects the *asymmetric* treatment of the space-time dependence of functions that we will use throughout the thesis.)

Gelfand Triple The solution theory of Evolution Problems is based on a "Gelfand triple" of Hilbert spaces. We introduce the general definition, comment on its existence and give a specific example, which we shall need in context of parabolic initial value problems.

Definition 1.2.12 (Gelfand Triple (Evolution Triple))
Let V and H be real Hilbert spaces such that $V \subset H$ is dense with a continuous embedding $J : V \hookrightarrow H$. Thus, there exists a constant $c_V > 0$ such that

$$\|\varphi\|_H \leq c_V \|\varphi\|_V \quad \text{for all } \varphi \in V. \tag{1.2}$$

Furthermore, we identify the space H with a subset of V^*, such that $H \subset V^*$ is dense, in particular, that there holds

$$V \hookrightarrow^J H \hookrightarrow^{J^*} V^*.$$

Then, we define a *gelfand triple* (or *Evolution Triple*) to be

$$(V, H, V^*).$$

Remark 1.2.13 (Existence of a Gelfand Triple)
The definition above is possible since for every element $u \in V$ there exist (anti-)linear functionals

$$v \mapsto (u, v)_V \quad \text{or} \quad v \mapsto (u, v)_H, \quad v \in V$$

in the dual space V^*. For more details refer to Lube 2007, Lemma 7.1.

Remark 1.2.14 ($H^1(\Omega)$ and $L^2(\Omega)$ Induce an Evolution Triple)
For a bounded domain Ω with Lipschitz-continuous boundary $\partial\Omega$ the spaces $V = H^1(\Omega)$ and $H = L^2(\Omega)$ induce an Evolution Triple.

Appropriate Spaces for Time-dependent Solutions As mentioned above, we introduce spaces of "abstract functions". According to Lube 2007, Section 7.3, on all these spaces a norm may be defined such that they become Banach spaces.
In a second step, we define respective spaces for integration and the actual "solution space" of the Evolution Problem (refer to Proposition 1.3.3). We will refrain from a careful introduction via Bochner integration theory (refer to Zeidler 1990, Chapter 23) but will define those spaces via completion of the continuous spaces in an appropriate norm.

Definition 1.2.15 (Spaces $C^m([a,b];X)$ of Abstract Functions)
Let $(X, \|\cdot\|_X)$ be a Hilbert space and $-\infty < a < b < \infty$. We call a vector-valued function $u : [a, b] \to X$ an *abstract function*.
We denote the space of all *continuous* abstract functions by $C([a, b]; X)$, where a vector-valued function u is *continuous* if $\lim_{\tau\to 0} \|u(t_0 + \tau) - u(t_0)\|_X = 0$ for all $t_0 \in [a, b]$.
The space of all m-times *differentiable* abstract functions shall be $C^m([a, b]; X)$, where an abstract function is called *differentiable* if for all $t \in [a, b]$ there exists $\tilde{u}(t) \in X$ with $\lim_{\tau\to 0} \|1/\tau(u(t + \tau) - u(t)) - \tilde{u}\|_X = 0$. We then write $u' = \tilde{u}$.

Definition 1.2.16 ($L^2(a,b;X)$, $H^m(a,b;X)$ and $W(0,T)$)
We define $L^2(a,b;X)$ to be the completion of $C([a,b];X)$ in the norm induced by the inner product

$$(u,v)_{L^2(a,b;X)} := \int_a^b (u(t),v(t))_X \,\mathrm{d}t.$$

Similarly, we define the space $H^m(a,b;X)$, $m \in \mathbb{N}$, to be the completion of $C^m([a,b];X)$. For $V \hookrightarrow H \hookrightarrow V^*$ a gelfand triple, we define $W(0,T)$ as the completion of $C^1([a,b];X)$ in the norm

$$\|u\|_{W(0,T)} = \|u\|_{L^2(0,T;V)} + \|u\|_{H^1(0,T;V^*)},$$

which leads to the following "representations" of the space $W(0,T)$:

$$W(0,T) = L^2(0,T;V) \cap H^1(0,T;V^*) = \{v \in L^2(0,T;V) \mid v' \in L^2(0,T;V^*)\}.$$

Defining Initial Values The following Lemma ensures, that it "makes sense" to impose *boundary conditions* on abstract functions in H, i.e. initial values for an Evolution Problem.

Lemma 1.2.17 ($W(0,T)$-functions may be be thought of to be continuous)
For $V \hookrightarrow H \hookrightarrow V^*$ a gelfand triple and $T > 0$ we obtain: $W(0,T) \subset C([0,T];H)$ with a continuous embedding.

Proof.
Refer to Lube 2007, Lemma 7.14 for instance. □

1.3 Evolution Equations of First Order

In this section, we introduce the problem for which we shall setup POD reduced-order models and carry out (sub)optimal control. We state the problem in two forms, provide respective ingredients and quote theoretical results on the solution. In the last subsection, "parabolic initial value problems", a special case of Evolution Problems, shall be investigated. This shall set the stage for practical applications, in particular to the non-stationary Heat Equation.

General Prerequisite Let $(V,(\cdot,\cdot)_V)$ and $(H,(\cdot,\cdot)_H)$ be real, *separable* Hilbert spaces with respective inner products. Suppose that V is *dense* in H with *compact embedding*: $V \hookrightarrow H$. Thus, we may construct a *Gelfand triple*

$$V \subset H \subset V^*.$$

Inner product in V Let the *inner product* in V as well as the *associated norm* be given by a bilinear form $a : V \times V \to \mathbb{R}$. Hence, for all $\varphi, \psi \in V$, there should hold

$$(\varphi,\psi)_V := a(\varphi,\psi) \quad \text{and} \quad \|\varphi\|_V := \sqrt{a(\varphi,\varphi)}. \tag{1.3}$$

In order to establish the solution theory, we need to require $a : V \times V \to \mathbb{R}$ to be *bounded and coercive*, i.e., there have to exist constants $M > 0$ and $\gamma > 0$ such that for all $u, v \in V$, there holds

$$a(u,v) \le M \|u\|_V \|u\|_V \quad \text{and} \quad a(v,v) \ge \gamma \|v\|_V^2.$$

1.3.1 Problem Statements

Let us now introduce the actual problem statement as well as its weak formulation.

Evolution Problem We consider an abstract parabolic Initial Value Problem (IVP) with *constant* coefficient in time. We also call it an IVP of an *evolution equation of first order*:

Problem 1.3.1 (Evolution Problem of First Order)
Let $L \in \mathcal{L}(V, V^*)$ be an *H-self-adjoint* operator. Furthermore, let $F \in L^2(0,T;V^*)$, $y_0 \in H$ and $0 < T < \infty$.
Then, the *Evolution Problem* of first order reads:

$$y'(t) + Ly(t) = F(t), \quad y(0) = y_0, \quad t \in (0,T]. \tag{1.4}$$

Generalized Problem Statement We wish to establish a "weak formulation" of the Evolution Problem. For this purpose, we specify the bilinear form a to be the bilinear form, which "corresponds" to the linear operator L:

$$a : V \times V \to \mathbb{R}, \quad a(u,v) := (Lu, v)_H, \quad \text{for all } u, v \in V.$$

Since L is assumed to be self-adjoint, a is symmetric:

$$a(u,v) := (Lu, v)_H = (u, L^*v)_H = (Lv, u)_H = a(v,u).$$

Problem 1.3.2 (Generalized Evolution Problem)
$y \in W(0,T)$ is called a *generalized solution* of the Evolution Problem (1.4) if there holds for $y_0 \in H$ and $t \in (0,T]$

$$\frac{d}{dt}(y(t), \varphi)_H + a(y(t), \varphi) = (F(t), \varphi)_H, \tag{1.5a}$$

$$y(0) = y_0, \tag{1.5b}$$

where (1.5a) may hold for all $t \in (0,T]$ and every test function $\varphi \in V$ in the sense of equality of functions in $L^2(0,T)$. Note that the requirement $y_0 \in H$ is possible due to Lemma 1.2.17.

1.3.2 Solution of Evolution Problems

We provide a result on the (unique) solvability of the system above and comment on the hence well-defined *data-solution operator*.

Associated Operator A We associate a linear operator A with a such that there holds:

$$\langle A\varphi, \psi \rangle_{V',V} = a(\varphi, \psi) \quad \text{for all } \varphi, \psi \in V.$$

Then, A is an *isomorphism* from V onto V'. Alternatively, A may be considered a *linear, unbounded, self-adjoint operator* in H with domain

$$D(A) := \{\varphi \in V : A\varphi \in H\} \qquad \text{and} \qquad D(A) \hookrightarrow V \hookrightarrow H = H^* \hookrightarrow V^*.$$

The chain of embeddings is true, since we have identified H with its dual H^*. In particular, all the embeddings are continuous and dense when $D(A)$ is endowed with the *graph norm* of A.

Existence and Uniqueness of Solution The following theorem guarantees the existence of a unique solution to Problem 1.3.2.

Proposition 1.3.3 (Unique Weak Solution)
Under the particular assumptions, Problem 1.3.2 admits a unique solution $y \in W(0,T)$: For every $f \in L^2(0,T;H)$ and $y_0 \in V$ there exists a unique weak solution of (1.5) satisfying

$$y \in C([0,T];V) \cap L^2(0,T;D(A)) \cap H^1(0,T;H). \tag{1.6}$$

Proof.
Refer to Dautray and Lions 1992, for instance. □

The Solution Operator and its Adjoint In context of control theory, we will make use of a *"solution operator"* $\mathcal{S}$ which maps the "data" $(f, y_0) \in L^2(0,T;V') \times H$ of the problem to the respective solution. Note that due to uniqueness of the solution for any choice of data, such an operator $\mathcal{S}$ is *well defined*.

Definition 1.3.4 (Solution Operator)
Define the solution operator $\mathcal{S}$ by

$$\mathcal{S} : L^2(0,T;V') \times H \to W(0,T), \quad y = \mathcal{S}(f,\varphi) \quad \text{such that } y \text{ solves (1.5)}$$

and the dual operator associated with it by

$$\mathcal{S}^* : W(0,T)' \to L^2(0,T;V) \times H$$

such that there holds

$$(w, \mathcal{S}(f,\varphi))_{W(0,T)',W(0,T)} = ((f,\varphi), \mathcal{S}^* w)_{L^2(0,T;V')\times H, L^2(0,T;V)\times H}$$

for all $(w, f, \varphi) \in W(0,T)' \times L^2(0,T;V') \times H$ and hence the definition of the adjoint is justified.

1.3.3 Parabolic Initial Value Problem of Second Order

We wish to apply the abstract theory of Section 1.3.2 to the case of an initial value problem (IVP) for parabolic partial differential equations of second order with *constant coefficients in time*. (These problems are quite common in physical applications.)
For a careful "classification" of partial differential equations refer to Lube 2007, Section 0.2. In this thesis, we investigate a specific parabolic IVP, which we derive from the (abstract) Evolution System (1.4) by choosing L to be a "differential operator of second order". Since we have required L to be self-adjoint, we may not include a "convective term" (which would induce an asymmetric additive term in the weak formulation of the problem). For coefficients to be specified, the operator L then reads:

$$L(y) = -\sum_{i,j=1}^{n} \frac{\partial}{\partial x_i}\left(a_{ij}\frac{\partial y}{\partial x_j}\right) + cy.$$

In particular, in this subsection we shall present a strong as well as a weak statement of the IVP and show its solvability. In Chapter 6, we will carry out numerical experiments for the *non-stationary Heat Equation* which is a special case of this type of parabolic IVP.

Strong PDE Statement In order to specify an IVP from a system of the general type of Evolution System (1.4), we additionally need to impose *boundary conditions* – in this context we restrict ourselves to the case of *homogeneous Dirichlet* boundary conditions.

Problem 1.3.5 (Parabolic IVP)
Let us make the following assumptions on the "data":

1. Let $\Omega \subset R^n$, $1 \leq n \in \mathbb{N}$, be a bounded domain with Lipschitz-continuous boundary $\partial\Omega$ and $0 < T < \infty$. Furthermore define $Q_T := (0,T) \times \Omega$.
2. The coefficients a_{ij} and c are independent of t. Furthermore, let be $a_{ij} = a_{ji}, c \in L^\infty(\Omega)$, $i,j = 1,\dots,n$, as well as $f \in L^2(Q_T)$ and $y_0 \in L^2(\Omega)$.
3. With a time-independent constant $\gamma > 0$ let be: $\sum_{i,j=1}^n a_{ij}(x)\xi_i\xi_j \geq \gamma |\xi|^2$ for all $x \in \Omega$ and $\xi \in \mathbb{R}^n$.
4. Let be $c(x) \geq 0$ for all $x \in \Omega$.

Then, the *parabolic Initial Value Problem* reads:

$$\begin{aligned} \frac{\partial y}{\partial t} - \sum_{i,j=1}^n \frac{\partial}{\partial x_i}\Big(a_{ij}\frac{\partial y}{\partial x_j}\Big) + cy &= f && \text{in} \quad Q_T, \\ y(t,x) &= 0 && \text{on} \quad (0,T)\times\partial\Omega, \\ y(0,x) &= y_0(x) && \text{in} \quad \{0\}\times\Omega. \end{aligned}$$

Weak Formulation Analogously to the abstract case, we introduce a generalized form of the problem. (For a detailed derivation of this statement refer to Lube 2007, Chapter 9 for instance.) Note that the choice $V = H_0^1(\Omega)$ is due to the boundary condition of Problem 1.3.5.

Problem 1.3.6 (Weak Formulation of Parabolic IVP)
Define the spaces

$$V := H_0^1(\Omega), \quad H := L^2(\Omega), \quad W(0,T) := \{y \in L^2(0,T;V) : y' \in L^2(0,T;V^*)\}.$$

Then, find $y \in W(0,T)$ such that for all $v \in V$

$$(y'(t),v)_H + a(y(t),v) = (F(t),v)_H\,, \tag{1.8a}$$
$$y(0) = y_0 \in H \tag{1.8b}$$

with

$$a(y,v) := \int_\Omega \Big(\sum_{i,j=1}^n a_{ij}(x)\frac{\partial y}{\partial x_j}\frac{\partial v}{\partial x_i} + c(x)yv\Big)\,\mathrm{d}x, \tag{1.9}$$

$$(F(t),v)_H := \int_\Omega f(t,x)v\,\mathrm{d}x. \tag{1.10}$$

Existence of a Solution which is Unique By means of Proposition 1.3.3, we may infer the following specialization to our problem of concern:

Proposition 1.3.7 (Existence/Uniqueness of IVP of 2nd order)
Under the assumptions of Problem 1.3.5, Problem 1.3.6 admits a solution $y \in W(0,T)$ which is uniquely determined.

Proof.
We need to show that all assumptions for Problem 1.3.2 are fulfilled. In detail, this

may be found in Lube 2007, Satz 9.1. □

1.4 Discretization of Evolution Problems

In order to actually *compute* a solution of Evolution Problems, we shall discretize the problem, i.e., we shall try to approximate the problem in a finite dimensional subspace. (The "Galerkin approach" we use is general. In particular, we shall also apply it to construct reduced-order models.)

Procedure We introduce the "vertical way" of discretization, i.e., we first approximate the problem in the space dimension and in a second step we discretize the resulting problem in time. We shall proceed on an "abstract" as well as on a "matrix" level. After that, we explain how to obtain a spatial approximation (of parabolic IVP) by means of the *Finite Element* method.

Horizontal vs Vertical Approach to Discretization In order to solve the Problem 1.3.2 numerically, we need to discretize the problem in space as well as in time. There are approaches that treat both these components simultaneously, i.e., perform a *full discretization*. Yet in order to apply the POD Method, we shall make use of a so-called *semi-discretization* which treats time and space separately.
A priori, there are two basic procedures: Discretize in time first and obtain a sequence of "stationary" problems each of which may be discretized in space. This procedure is called *horizontal method*.
On the other hand, we may first discretize in space and obtain a system of ordinary differential equations (ODEs) which then may be discretized in time. This so-called *vertical method* shall be applied in this context since we aim to significantly reduce the size of this ODE system by means of the POD Method (refer to Chapter 4 on "Reduced-Order Modeling").

1.4.1 Discretization in Space – Ritz-Galerkin Approach

We wish to "semi-discretize" the Problem 1.3.2 in space and make use of a so-called "Galerkin ansatz".

Galerkin Ansatz According to Proposition 1.3.3, Problem 1.3.2 admits a solution $y \in W(0,T)$. Therefore, $y(t)$ lies in V for each $t \in [0,T]$. In order to obtain our "semi-discretized" problem we make the ansatz to approximate V by a *finite dimensional* space, spanned by "Galerkin ansatz functions" $\{\varphi_i\}_{i=1}^q$:

$$V_h := \operatorname{span}(\varphi_1, \dots, \varphi_q) \subset V, \quad q \in \mathbb{N}.$$

A suitable initial value $y_0^h \in V_h$ may be obtained by an $(\cdot,\cdot)_H$-orthogonal projection of $y_0 \in H$ on V_h, for instance. The semi-discrete problem statement then reads:

Problem 1.4.1 (Semi-Discrete Initial Value Problem)
Find $y_h \in L^2(0,T;V_h)$ with $y_h' \in L^2(0,T;V_h^*)$, such that for an appropriate initial value $y_0^h \in H$, there holds

$$(y_h(t), v)_H + a(y_h(t), v) = (F(t), v)_H \quad \text{for all } v \in V_h, \tag{1.11a}$$

$$y_h(0) = y_0^h \in H. \tag{1.11b}$$

Semi-discrete System of "Ordinary" Initial Value Problems Using the respective basis of the finite-dimensional spaces involved, it is sufficient for the solution of (1.11) to determine the respective coefficients. This procedure essentially reduces to the solution of a linear system of "ordinary differential equations" (which in general is *considerably large*).

Proposition 1.4.2 (System of Ordinary Initial Value Problems)
Let us make the ansatz

$$y_h(t) := \sum_{j=1}^{q} c_j(t)\varphi_j, \qquad y_h(0) = \sum_{j=1}^{q} \alpha_j \varphi_j \tag{1.12}$$

and introduce the *matrices* and *vectors*

$$\begin{aligned} D &:= ((\varphi_j, \varphi_i)_H)_{i,j=1}^q, \quad A := (a(\varphi_j, \varphi_i))_{i,j=1}^q, \\ \tilde{F}(t) &:= ((F(t), \varphi_i)_H)_{j=1}^q, \quad \tilde{g} := (\alpha_j)_{j=1}^q. \end{aligned} \tag{1.13}$$

Then, we may obtain the coefficients $\tilde{c}(t) := (c_j(t))_{j=1}^q \in \mathbb{R}^q$, $t \in [0,T]$, of the solution y_h of (1.11) from the following "*ordinary initial value problem*" of first order (for $t \in (0,T]$):

$$D\frac{d}{dt}\tilde{c}(t) + A\tilde{c}(t) = \tilde{F}(t), \quad \tilde{c}(0) = \tilde{g}. \tag{1.14}$$

Proof.
Inserting the ansatz (1.12) and successively choosing $v = \varphi_i, i = 1, ..., n$ in (1.11), we arrive at

$$\sum_j (\varphi_j, \varphi_i)_H \frac{d}{dt} c_j(t) + \sum_j a(\varphi_j, \varphi_i) c_j(t) = (F(t), \varphi_i)_H, \quad i = 1, \ldots, n \tag{1.15a}$$

$$c_j(0) = \alpha_j, \quad j = 1, \ldots, n, \tag{1.15b}$$

which immediately gives system (1.14) (by using the respective definitions). □

1.4.2 Discretization in Time – Theta Scheme

We now wish to discretize Problem 1.4.1 in time as well. We make use of the so-called *one step θ-scheme* (refer to Lube 2007, Chapter 6 for details).

Definitions Let $\Lambda = \{t_m\}_{m=0}^M$ be a partition of the time interval of consideration $[0,T]$ with $t_0 := 0$, $t_M := T$ and $\tau_m := t_{m+1} - t_m$ being the "time step size". Furthermore, let be $I_m := (t_m, t_{m+1})$ and the parameter $\theta \in [0,1]$. Then, define:

$$\begin{aligned} Y_m &:= Y(t_m), & Y_{m+\theta} &:= \theta Y_{m+1} + (1-\theta) Y_m, \\ F_m &:= F(t_m), & F_{m+\theta} &:= \theta F_{m+1} + (1-\theta) F_m. \end{aligned}$$

Let us also introduce the matrix $\hat{F}$ of all right-hand sides $\tilde{F}(\tau_m)$ at the respective time instances $\tau_m \in \Lambda$:

$$\hat{F} = \left(\tilde{F}(t_0), \ldots, \tilde{F}(t_M)\right) = \left(((F(t_0), \varphi_j)_H)_{j=1}^q, \ldots, ((F(t_M), \varphi_j)_H)_{j=1}^q\right). \tag{1.16}$$

Fully Discrete Problem and Solution We may now state the fully discrete problem and provide a result on its solution.

Problem 1.4.3 (Fully Discrete Problem)
Find $Y_{m+1} \in V_h$, $m = 0, \dots, M-1$, such that

$$\left(\frac{Y_{m+1} - Y_m}{\tau_m}, v\right)_H + a(Y_{m+\theta}, v) = (F_{m+\theta}, v)_H\,, \qquad v \in V_h, \tag{1.17a}$$

$$(Y_0, w)_H = (y_0, w)_H\,, \qquad w \in V_h. \tag{1.17b}$$

Lemma 1.4.4 (Existence and Uniqueness of Solution)
For a V-coercive bilinear form $a(\cdot,\cdot)$ and for $\max_{m=0,\dots,M}\ \tau_m$ sufficiently small, Problem 1.4.3 admits a unique solution.

Proof.
According to (1.17), the solution to Problem 1.4.3 corresponds to the successive approximation to the solution of the following variational problem (for a suitable right-hand side G_m, $m = 0, \dots, M-1$):

$$\theta a(Y_{m+1}, v) + \frac{1}{\tau_m}(Y_{m+1}, v)_H = (G_{m+1}, v)_H\,.$$

For a V-coercive bilinear form $a(\cdot,\cdot)$ and $\max_{m=0,\dots,M}\tau_m$ sufficiently small, existence and uniqueness of a solution may then be deduced from Lemma 1.2.11 (Lax-Milgram) since

$$\begin{aligned}\theta a(v,v) + \frac{1}{\tau_m}(v,v)_H &\geq \theta(\gamma \|v\|_V^2 - \delta \|v\|_H^2) + \frac{1}{\tau_m}\|v\|_H^2) \\ &\geq \theta\delta \|v\|_V^2 - \left(\frac{1}{\tau_m} - \delta\theta\right)\|v\|_H^2\,.\end{aligned}$$

□

The Implicit Euler Scheme on the Level of Matrices In analogy to the space discretization, we wish to establish a formulation of the time-discretization on a "matrix level". For that purpose, we may either use the θ-scheme for (1.14) or use a basis ansatz for (1.17) (as we did in the derivation of (1.14)).
Important special cases of the θ-scheme discretization are the *explicit Euler method* for $\theta = 0$, the *Crank-Nicolson method* for $\theta = 1/2$ and the *implicit Euler method* for $\theta = 1$. In order to simplify the presentation, we shall concentrate on the latter example which we shall use throughout the thesis.
For $0 \leq m \leq M$, we additionally introduce the notation

$$C^m := (c_j(t_m))_{j=1}^q \in \mathbb{R}^q \quad \text{and} \quad \tilde{F}^m := \tilde{F}(t_m).$$

We choose $\theta = 1$ in (1.17) and make use of a basis ansatz such as in (1.12). Then, the implicit Euler scheme for (1.14) yields

$$D\frac{C^{m+1} - C^m}{\tau_m} + AC^{m+1} = \tilde{F}^{m+1}, \quad 0 \leq m \leq M-1.$$

Multiplying be τ_m and rearranging, we obtain:

$$\underbrace{(D + \tau_m A)}_{:=P_m} C^{m+1} - DC^m = \tau_m \tilde{F}^{m+1}, \quad 0 \leq m \leq M-1.$$

We way now summarize this "system of equations in $\mathbb{R}^q$" by gathering all coefficients C^m in one vector and building a corresponding *block* matrix of size $qM \times qM$. (This is not necessarily efficient,

yet simple in terms of implementation.) Together with the initial value $\tilde{g}$ we then obtain the system (all entries which are not specified are assumed to be zero and I denotes the identity of size q):

$$\begin{pmatrix} I & & & & \\ -D & P_0 & & & \\ & -D & P_1 & & \\ & & \ddots & \ddots & \\ & & & -D & P_{M-1} \end{pmatrix} \begin{pmatrix} C^0 \\ C^1 \\ C^2 \\ \vdots \\ C^M \end{pmatrix} = \begin{pmatrix} \tilde{g} \\ \tau_1 \tilde{F}^1 \\ \tau_2 \tilde{F}^2 \\ \vdots \\ \tau_M \tilde{F}^M \end{pmatrix}. \tag{1.18}$$

1.4.3 Spatial Approximation of parabolic IVP by Finite Elements

In Subsection 1.3.3, we have found that the solution space of choice for the weak formulation of the parabolic IVP (Problem 1.3.6) is $V = H_0^1(\Omega)$. According to the previous subsection, we hence need to find a finite dimensional approximation V_h of $H_0^1(\Omega)$. Therefore, we shall construct a suitable space V_h and also provide a basis for it.
We may then find a solution to the IVP by choosing the ansatz functions φ in the system of ordinary IVPs of Proposition 1.4.2 to be FE ansatz functions and calculating the respective coefficients $\tilde{c}$.

Idea of the FE Method The Finite Element Method is a special method to construct a *finite dimensional Hilbert Space* to approximate an infinite dimensional Hilbert Space X.
Furthermore, we desire the basis function to have a rather small support ("local basis") in order to have only few "couplings" amongst these functions. Then, their inner product is zero nearly everywhere and hence the matrix A in (1.13) is sparse. This shall in turn save memory and fasten the solution of the resulting system in Proposition 1.4.2.

Triangulation of the Domain Taking up on the idea of a "local basis", we introduce a "decomposition" of the domain Ω to construct "Finite Elements" on. (We require Ω to be polyhedral in order to be able to *exactly* decompose the domain in the fashion proposed.)

Definition 1.4.5 ((Admissible) Triangulation)
Let Ω be a bounded, polyhedral domain in $\mathbb{R}^n$, $n = 1, 2, 3$, and let $\{h\}_{h>0}$ be a family with accumulation point zero. Then, the family $\{\mathcal{T}_h = \{K_i\}_{i=1}^M\}_{h>0}$ of non-overlapping decompositions of Ω into convex, polyhedral subdomains K_i which satisfy

$$\overline{\Omega} = \bigcup_{j=1}^{M} K_j \qquad \text{and} \qquad K_i \cap K_j = \varnothing \quad \text{for} \quad i \neq j$$

is called a family of *triangulations*. We set $h_i := \operatorname{diam}(K_i)$ and $h := \max_{i=1,\dots,M} h_i$.
A triangulation $\mathcal{T}_h$ is called *admissible* if two different $K_j, K_i \in \mathcal{T}_h$ are either pairwise disjoint or have exactly one whole face (only for $n = 3$), edge (only for $n \geq 2$) or vertex in common.
We may describe Ω by the set of all $\overline{N}$ vortices $\{p^i\}_{i=1}^{\overline{N}}$ in $\mathcal{T}_h$ and may define each individual subdomain $K \in \mathcal{T}_h$ as the "convex hull" of the vortices belonging to K.

Abstract Finite Elements A "Finite Element" (FE) is a triple $(K, \mathcal{P}, \Sigma)$, consisting of a convex polyhedral domain $K \subset R^n$, a finite dimensional linear space $\mathcal{P}$ of functions defined on K as well as a basis Σ of the dual space K^* of K which is also called the set of *degrees of freedom*.
We characterize Finite Elements by the type of subdomains K, the "ansatz space" $\mathcal{P}$ and the set of functionals Σ, i.e., the location and type of the degrees of freedom.

Linear Triangular Lagrange-elements For $\Omega \subset R^n, n = 1, 2$, let us now construct the certain class of Finite Elements that we shall use in the numerical experiments in Chapter 6.
Let $\mathcal{T}_h = \{K_i\}_{i=1}^M$ be an admissible triangulation of the domain $\Omega \subset R^2$ in convex polyhedral subdomains $K_i \in \mathcal{T}_h$.
The ansatz space $\mathcal{P}$ shall consist of *piecewise linear ansatz functions* over K. The degrees of freedom shall coincide with the vortices. (In analogy to polynomial interpolation, these elements are then called Lagrange elements.)
For that purpose, we construct the set $P_1(K)$ of affine functions $\{\varphi_j\}_{j=1}^3$ over K with the property $\varphi_j(p^k) = \delta_{jk}$. (Technically, an easy way to do that is to make use of so-called *Barycentric coordinates.*)
Similarly, we could proceed for other types and locations of degrees of freedom, higher order ansatz functions, higher space dimensions or different types of subdomains (such as rectangles).

Obtaining an FE Space We now wish to construct a "global" function space on Ω by gathering all the "local" Finite Elements. In order to do that, it is helpful to "parameterize" the subdomains $K \in \mathcal{T}_h$: We choose an "independent" reference element $\hat{K}$ and then obtain all other elements K by means of a multi-linear mapping $F_K : \hat{K} \to K$. This mapping shall in particular map all vortices of $\hat{K}$ to vortices of K without permutations.
Apart from that, we have not considered properties of the resulting "global" functions yet: For instance, if for an FE space V_h, there holds $V_h \subset C(\overline{\Omega})$, we call V_h a C^0*-FE space.*
By means of the "parametrization" F we may now construct such a C^0-FE space for the choice of ansatz spaces $P_l, l \in \mathbb{N}$, introduced above:

Proposition 1.4.6 (Lagrange FE Space)
Let $\mathcal{T}$ be an admissible triangulation of the bounded, polyhedral domain $\Omega \subset \mathbb{R}^n$ in regular simplices.
Then, the Lagrange-elements of class $P_k, k \in \mathbb{N}$ built a C^0-Finite-Element-Space $X_\mathcal{T}$. In particular, for

$$X_\mathcal{T} := \{u \in C(\overline{\Omega}) \mid u|_K \circ F_K \in \mathcal{P}_j(\hat{K}), K \in \mathcal{T}_h \text{ and } u = 0 \text{ on a neighborhood of } \partial\Omega\}$$

there holds

$$X_\mathcal{T} \quad \subset \quad H_0^1(\Omega).$$

Proof.
For the case $n = 2$, refer to Brenner and Scott 2002, Satz 3.3.17 and for the subspace result, refer to Knabner and Angermann 2000, Satz 3.20. □

Convergence Properties of Spaces Proposition 1.4.6 implies that $V_h := X_\mathcal{T}$ yields a conform approximation of $V = H_0^1(\Omega)$ as desired in the introduction of this subsection. (Refer also to Knabner and Angermann 2000, Satz 3.23.)
We should note that V_h actually "converges" to V for a decreasing size of the triangulation h. This holds true since for the continuous solution $u \in V$ and a constant C there holds:

$$\|u - u_h\|_{L^2(\Omega)} \leq Ch\,|u|_1\,. \tag{1.19}$$

A "general" version of (1.19), depending on the regularity assumed for the solution u, is given in Knabner and Angermann 2000, Satz 3.29. For example, for $u \in H^2(\Omega)$, we obtain

$$\|u - u_h\|_{H^1(\Omega)} \leq Ch\,|u|_2\,.$$

Characterization of the FE Space $X_\mathcal{T}$ Let us now show that the FE space $X_\mathcal{T}$ as a finite dimensional Hilbert space is isomorphic to $\mathbb{R}^q$ with a suitable inner product. In order to define those

products, we introduce "FE matrices". Finally, we define an FE vector in order to conveniently denote the "coefficients with respect to a Finite Element basis".

Definition 1.4.7 (Mass and stiffness matrices)
Let $\mathcal{B}(V_h) = \{\varphi_i\}_{i=1}^q$ be a basis of an FE space V_h. Then, define the *mass matrix* M and the *stiffness matrix* S "entry-wise" as

$$M_{ij} := \int_\Omega \varphi_i(x)\varphi_j(x)\,\mathrm{d}x \quad \text{and} \quad S_{ij} := \int_\Omega \varphi_i(x)\varphi_j(x) + \nabla\varphi_i(x)\nabla\varphi_j(x)\,\mathrm{d}x.$$

Proposition 1.4.8 ($X_\mathcal{T}$ is isomorphic to $\mathbb{R}^q$)
For q the number of degrees of freedom, the FE-Space $X_\mathcal{T}$ is isomorphic to $\mathbb{R}^q$. Endowing them with the inner products, we may interpret $X_\mathcal{T}$ as "discrete analogues" $L_h^2(\Omega)$ and $H_h^1(\Omega)$ of $L^2(\Omega)$ and $H^1(\Omega)$, respectively.

$$(u,w)_{L_h^2(\Omega)} = (u, Mv)_{\mathbb{R}^q} \quad \text{or} \quad (u,w)_{H_h^1(\Omega)} = (u, Sv)_{\mathbb{R}^q},$$

where M and S denote the mass- and the stiffness matrix, respectively.

Proof.
Since $X_\mathcal{T}$ is a q-dimensional vector space, it naturally is isomorphic to $\mathbb{R}^q$.

• Basis Representation We may set up a basis $\mathcal{B} = \{\varphi_i\}_{i=1}^q$ of $X_\mathcal{T}$ and represent any element $u \in X_\mathcal{T}$ by a coefficient vector $U \in \mathbb{R}^q$ since we may write

$$u = \sum_{k=1}^q U^{(k)}\varphi_k. \tag{1.20}$$

• L^2-Inner Product Let us now show that the L^2-inner product of two functions $u, v \in X_\mathcal{T}$ induces the product $(\cdot,\cdot)_{L_h^2(\Omega)}$. By means of (1.20), we obtain

$$\begin{aligned}(u,v)_{L^2(\Omega)} &= \int_\Omega u(x)v(x)\,\mathrm{d}x = \sum_{j=1}^q\sum_{k=1}^q U^{(j)}\int_\Omega \varphi_j(x)\varphi_k(x)\,\mathrm{d}x V^{(k)} \\ &= \sum_{j=1}^q\sum_{k=1}^q U^{(j)}M_{jk}V^{(k)} = U^T MV = (u,v)_{L_h^2(\Omega)}.\end{aligned}$$

• H^1-Inner Product The proof for $(\cdot,\cdot)_{H_h^1(\Omega)}$ is perfectly analogous to the L^2 case. □

Definition 1.4.9 (FE vector)
The vector $U \in \mathbb{R}^q$ of all coefficients of a function u_h in its expansion in terms of a FE basis as in (1.20) is called (the corresponding) *FE vector*.

Chapter 2

The POD Method in Hilbert Spaces

In this chapter, we introduce the *Proper Orthogonal Decomposition* (POD) Method. We wish to focus on its idea, phrase it mathematically and thoroughly investigate it from various points of view – where the stress shall lie on "abstract" results in order to be able to refine them for specific cases in later chapters.
Thus, also the "context" of the POD Problem shall be chosen to be the most general one which is useful for the main application of the method in this thesis: constructing *low-order models* for Evolution Systems.
For that purpose, we concentrate on "abstract ensembles", lying in a separable Hilbert space X, which are "parameterized" over a real interval. Thus, assume that X denotes a separable Hilbert space throughout this chapter.

Relation to the Other Chapters on POD In Chapter 3, the "abstract" ensemble shall be obtained from the solution of an evolution problem, parameterized over (naturally *real*) values of time.
In Chapter A, we shall further investigate the method itself and give a few insides in what happens "in the background". Furthermore, perfectly different applications of the POD Method are enlightened ("Decomposition of Signals"; "Finding Coherent Structures").

Procedure We motivate the POD Problem and define it in "abstract" fashion. Then, we show the existence of a solution and provide characterizations of it. Finally, we investigate the behaviour of the POD Basis when "discrete" ensembles converge to "continuous" ones.

Literature In context of the Lagrangian case and the error analysis, the argumentation roughly follows Holmes, Lumley, and Berkooz 1996 as well as Volkwein 2001*b* and in terms of asymptotic analysis we follow Kunisch and Volkwein 2002. Especially the basic concepts are following the basic book Holmes, Lumley, and Berkooz 1996.

2.1 The Abstract POD Problem

Let us start by defining the essential notation and stating the POD Problem. (The *orthogonality* in the problem definition has given the method its name.) Note that $\|\cdot\|_2$ denotes the 2-norm in $\mathbb{R}^\ell$.

The Practical Objective Recall that we wish to construct an "intelligent" basis for an Evolution System in order to use it for a ROM (refer to the introduction). The POD Method is useful in this context as it, roughly speaking, aims to extract the *essential ingredients* of a given data ensemble $\mathcal{V}$; it tries to find *typical* elements of $\mathcal{V}$.

The Formal Objective In other words, POD aims to determine the (on an average $\langle\cdot\rangle_{t\in\Gamma}$) optimal representation of some order ℓ of $\mathcal{V}$ – optimal in the sense that there is no other method giving a better approximation of the same rank or lower.
Hence, in a more formal way our goal is to establish an (*orthonormal*) basis $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ for a subspace $\mathcal{V}^\ell \subset \mathcal{V}$ such that $\mathcal{V}^\ell$ is an optimal "space-representation" for $\mathcal{V}$. It remains to define such a "representation of spaces", which we shall carry out element-wise: As a representation of $g \in \mathcal{V}$ we choose the *best approximation* g^ℓ of g by elements of $\mathcal{V}^\ell$.
We introduce a norm $\|\cdot\|_X$ in order to define the error of this representation to be $\left\|g - g^\ell\right\|_X$. We now desire the average (over $\mathcal{V}$) of the square of all these errors to be minimal.
Note that alternatively to minimizing the error, we may try to maximize – on average – the contribution of the POD Basis elements to the ensemble $\mathcal{V}$. (The equivalence of the two statements is shown in Proposition 2.1.6.)

2.1.1 Ingredients for the POD Problem Definition

Let us carefully define all the ingredients of the "formal objective" in order to define a "POD Problem" mathematically.

The Ensemble As mentioned above, we wish to optimally represent an ensemble $\mathcal{V}$. For technical reasons, let us define some "language" in this context. For example, in order to carry out the averaging operation conveniently, it is helpful to establish a parametrization y of the ensemble, which should be square-integrable for the average operation to be well-defined. For this purpose, we also define an "ensemble parameter set", that we only require to be a compact real interval Γ. – Of course more general assumptions are possible (in case of parameter estimation problems for example). Yet as we wish to apply the method to Evolution Problems, this parameter shall represent "time" which of course fulfills this requirement. In the remainder, we shall also need the property that the parametrization is continuous if Γ is not discrete. Let us summarize all of these necessities in the following definition:

Definition 2.1.1 (Ensemble Grid, Ensemble Set and Ensemble Space)
Let X be a separable Hilbert space and the *ensemble* $\mathcal{V}_P \subset X$ a set of elements in X. We then define the *ensemble space* to be $\mathcal{V} := \operatorname{span}(\mathcal{V}_P)$.
We endow X with an inner product $(\cdot,\cdot)_X$ such that the induced norm $\|\cdot\|_X$ measures a feature of $\mathcal{V}$ which we desire to represent well.
We furthermore introduce an *ensemble parameter set*, which shall either be a real interval Γ or a discrete *"ensemble grid"* $\Gamma_n \subset \mathbb{R}$. Assume that the elements of $\mathcal{V}_P$ may be parameterized over the ensemble parameter set by a (bijective) parameterization:

$$y \in L^2(\Gamma, \mathcal{V}_P).$$

Let us conveniently denote this *"parametrized ensemble"* as $(\Gamma, y, \mathcal{V}_P)$. For a continuous ensemble parameter set Γ, we additionally require (refer to Proposition 2.2.8)

$$y \in C(\Gamma; V).$$

For a discrete ensemble grid $\Gamma_n = \{t_j\}_{j=0}^n$, we define

$$\delta t := \min\{\delta t_j : 1 \le j \le n\} \quad \text{and} \quad \Delta t := \max\{\delta t_j : 1 \le j \le n\},$$

where $\delta t_j := t_j - t_{j-1}$ for $j = 1, \dots, n$.

Best Approximation For any choice of the inner product $(\cdot,\cdot)_X$, we may conclude that the best approximation $g^\ell \in \mathcal{V}^\ell$ of $g \in \mathcal{V}$ is given by the $(\cdot,\cdot)_X$-orthogonal projection of g onto $\mathcal{V}^\ell$ (refer to

Lube 2005, Satz 8.16). Since $\mathcal{V}^\ell$ is of finite dimension and $\mathcal{B}^\ell$ is assumed to be orthonormal, we may (by Definition 1.2.2) denote this projection in Fourier form:

Definition 2.1.2 (POD Projection)
For $\mathcal{V}$ and X of Definition 2.1.1 and an *orthonormal* basis $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell \subset \mathcal{V}$ of $\mathcal{V}^\ell$, we define the (orthogonal) *POD Projection* to be given by

$$P^\ell : \mathcal{V} \to \mathcal{V}^\ell, \quad P^\ell y = \sum_{k=1}^{\ell} (y, \psi_k)_X \, \psi_k. \tag{2.1}$$

Average Operator Roughly speaking, we define an average operator as the inner product in the parametrization space $L^2(\Gamma, \mathbb{R})$ of all elements of an ensemble set $\mathcal{V}_P$ with a weighting function $\omega \in L^2(\Omega)$.

Definition 2.1.3 (Average Operator)
For a parametrized ensemble $(\Gamma, y, \mathcal{V}_P)$, a *weighting function* $\omega \in L^2(\Omega)$ and an appropriate measure $\mathrm{d}t$ on Γ, we define an *average operator* over the corresponding ensemble space $\mathcal{V}$ to be

$$\langle \omega, \cdot \rangle_{\mathcal{V}_P} : L^2(\Gamma, \mathcal{V}_P) \to \mathcal{V}_P, \quad \langle \omega, y \rangle_{\mathcal{V}_P} := (\omega, y)_{L^2(\Gamma)} = \int_\Gamma \omega(t) y(t) \, \mathrm{d}t.$$

Note that for the discrete set $\Gamma_n = \{t_j\}_{j \in \mathbb{N}}$ we set (using a "counting measure" $\mathrm{d}t$)

$$(\omega, y)_{L^2(\Gamma_n)} := \sum_{j \in \mathbb{N}} \omega(t_j) y(t_j). \tag{2.2}$$

Averaging Functional Values In the remainder, we actually aim to average values of functionals $F : \mathcal{V} \to Z$ in a Hilbert space Z over $\mathcal{V}$. (The actual definition of Z shall be implied by the context and usually is either $Z := \mathbb{R}$ or $Z := X$.) In order to simplify notation, we introduce an *abbreviated notation* for the average operator introduced above. In particular, we do not explicitly denote the weighting function ω. Note that for $F \in L^2(\mathcal{V}, Z)$, we have $F \circ y \in L^2(\Gamma, Z)$, since $\mathrm{Im}(y) = \mathcal{V}_P \subset \mathcal{V}$. So we introduce:

Definition 2.1.4 (Simplified, Extended Average Operator)
Let Z be a Hilbert space and $F \in L^2(\mathcal{V}, Z)$. Define the "abbreviated notation" of the average operator $\langle \omega, \cdot \rangle_{\mathcal{V}_P}$ to be

$$\langle \cdot \rangle_{t \in \Gamma} : L^2(\Gamma, Z) \to Z, \quad \langle F \circ y(t) \rangle_{t \in \Gamma} := \langle \omega, F \circ y(t) \rangle_{\mathcal{V}_P}.$$

Illustration of the Simplified Average Operator In order to enlighten the "simplification" of the average operator, let us give an example:
For the discrete ensemble $(\Gamma_n = \{t_j\}_{j \in \mathbb{N}}, y^n, \mathcal{V}_P^n)$ and the continuous ensemble $(\Gamma = [0, T], y, \mathcal{V}_P)$, we obtain for $F(y(t)) := y(t) \, \|y(t)\|_X$ (and $y_j := y^n(t_j)$):

$$\langle y^n(t) \, \|y^n(t)\|_X \rangle_{t \in \Gamma_n} = \sum_{j \in \mathbb{N}} y^n(t_j) \, \|y^n(t_j)\|_X = \sum_{j \in \mathbb{N}} y_j \, \|y_j\|_X,$$

$$\langle y(t) \, \|y(t)\|_X \rangle_{t \in \Gamma} = \int_0^T y(t) \, \|y(t)\|_X \, \mathrm{d}t.$$

2.1.2 Statements of a POD Problem

Having gathered and discussed all ingredients necessary, we may now formulate a "POD Problem" mathematically. We introduce a "best approximation" version as well as "highest contribution" form and show that the statements are equivalent. By means of the two statements, we shall additionally give a remark on the choice of the norm $\|\cdot\|_X$.

Best Approximation Statement Let us now summarize all these findings in the following definition, such that the "mean square error" in the best approximations (over the Ensemble Space) is minimal (see above).

Definition 2.1.5 (POD Problem, POD Basis, POD Mode)
Fix $\ell \in \mathbb{N}$. Let $(\Gamma, y, \mathcal{V}_P)$ be a parameterized ensemble in a separable Hilbert space $(X,(\cdot,\cdot)_X)$. Let $\langle\cdot\rangle_{t\in\Gamma}$ be an *average operator* over the corresponding ensemble space $\mathcal{V}$, that commutes with any POD Projection P^ℓ.
Then, an *orthonormal* basis $\mathcal{B}^\ell = \{\psi_i\}_{i=1}^\ell$ of the *ℓ-dimensional subspace* $\mathcal{V}^\ell \subset X$ is called a *POD Basis* of rank ℓ if it fulfills the *POD Problem* (in "Best Approximation" version)

$$(\Gamma, y, \mathcal{V}_P, P^\ell, \langle\cdot\rangle_{t\in\Gamma}), \qquad \min_{\mathcal{B}^\ell} \left\langle \left\| y(t) - P^\ell y(t) \right\|_X^2 \right\rangle_{t\in\Gamma}, \tag{2.3}$$

where $P^\ell : \mathcal{V} \to \mathcal{V}^\ell$ denotes the *POD Projection*, belonging to $\mathcal{B}^\ell$. The elements of the POD Basis, the POD Basis vectors, we call *POD modes*.

Highest Contribution Formulation Note that the previous definition is intuitive in terms of the idea of the method, yet the following (equivalent) statement of the problem will turn out to be more handy when it comes to investigating the problem mathematically:

Proposition 2.1.6 (Highest Contribution Form of POD Problem)
The POD Basis may also be obtained from the *Alternative POD Problem*

$$(\Gamma, y, \mathcal{V}_P, P^\ell, \langle\cdot\rangle_{t\in\Gamma}), \qquad \max_{\mathcal{B}^\ell} \left\langle \left\| \left((\psi_i, y(t))_X \right)_{i=1}^\ell \right\|_2^2 \right\rangle_{t\in\Gamma}, \tag{2.4}$$

even though the respective extremal values differ.

Proof.
For the sake of brevity of notation, fix $t \in \mathcal{V}$ and denote $g = y(t)$. With the representation of the POD Projection (2.1), let us investigate the approximation version and

show that the two statements lead to the same POD Basis (t is fixed, but arbitrary):

$$\begin{aligned}
&\left\| g - \sum_{k=1}^{\ell} (g, \psi_k)_X \, \psi_k \right\|_X^2 \\
&= (g,g)_X - 2\left(g, \sum_{k=1}^{\ell} (g,\psi_k)_X \, \psi_k \right)_X + \left(\sum_{k=1}^{\ell} (g,\psi_k)_X \, \psi_k, \sum_{k=1}^{\ell} (g,\psi_k)_X \, \psi_k \right)_X \\
&= (g,g)_X - 2\sum_{k=1}^{\ell} |(g,\psi_k)_X|^2 + \sum_{k=1}^{\ell} |(g,\psi_k)_X|^2 \underbrace{(\psi_k,\psi_k)_X}_{=1} \\
&= \|g\|_X^2 - \sum_{k=1}^{\ell} |(g,\psi_k)_X|^2 \\
&= \|g\|_X^2 - \left\| \left((\psi_i, g)_X \right)_{i=1}^{\ell} \right\|_2^2 .
\end{aligned}$$

As $g = y(t)$ is given by the ensemble parameterization y, the approximation term of course is minimal if the second summand is maximal and vice versa. These precisely are the two statements of the POD Problem. As claimed, the two extremal values do not coincide. □

Relation of Optimality Norm Let us justify the choice of inner product in Definition 2.1.5. For this purpose, let us enlighten that the "Optimality Norm" (the norm in which the POD representation is optimal) is actually given by the norm which is induced by the inner product in X. Hence this norm should "measure a feature which we desire to represent well". In case the ensemble comes from a simulation of (say) a fluid flow, that "feature" could be the energy or the vorticity of the flow for example.

Remark 2.1.7 (Relation of Optimality Norm and POD Projection)
The "Optimality Norm" coincides with the norm which is induced by the inner product that defines the POD projection.
This fact is due to the following considerations: Choosing an inner product $(\cdot,\cdot)_X$ in X leads to the Fourier representation (2.1) of the $(\cdot,\cdot)_X$-orthogonal projection in any orthogonal basis. We are then in the position to solve problem (2.4).
On the other hand, if we wish to interpret the POD Basis as the "on average best approximation of the ensemble $\mathcal{V}$ of a certain rank", we should use the other statement (2.3) which technically just states this fact. Furthermore, it immediately follows form the problem statement that the optimal representation is given in the norm $\|\cdot\|_X$, which is induced by the inner product.
Proposition 2.1.6 teaches us that these formulations of a POD Problem are equivalent and hence the "assertion" follows.

2.2 Solution of Abstract POD Problems

In this section, we shall show the existence of a solution to the abstract POD Problem of Definition 2.1.5 and comment on the estimation the error of the representation.
In particular, we characterize it as orthonormal eigenvectors of two different operators, the "classical" one being (refer to Theorem 2.2.3):

Definition 2.2.1 (POD Operator)
For the Hilbert space X and the "parameterized ensemble" $(\Gamma, y, \mathcal{V})$ of Definition 2.1.5,

we define the *POD Operator* R to be

$$R : X \to \mathcal{V}, \quad R\psi = \langle y(t)\,(y(t),\psi)_X \rangle_{t\in\Gamma} \quad \text{for all} \quad \psi \in X. \tag{2.5}$$

Procedure We first give an idea of a necessary condition by reducing the problem to just one POD Basis element and applying the Lagrangian method for constrained optimization problems. We complete the characterization by showing that the necessary condition found is also sufficient and giving an error estimate.
We show that there exists a solution to the characterization (which implies the existence of a POD Basis). Finally, we give an alternative characterization of the POD Basis in anticipation of the practical applications in Chapter 3.

2.2.1 Motivation: Preliminary Necessary Optimality Condition

As setting up the full necessary optimality condition will become quite technical, we wish to give an idea of the conditions of the simpler problem of finding just a single POD Basis vector:

$$\max_{\psi} \left\langle |(y(t),\psi)_X|^2 \right\rangle_{t\in\Gamma} \quad \text{s.t.} \quad \|\psi\|_X^2 = 1 \tag{2.6}$$

(For the case $X = L^2([0,1])$ this might be found in Holmes, Lumley, and Berkooz 1996, section 3.1.) In this context we obtain:

Proposition 2.2.2 (Preliminary Necessary Condition for a POD Basis)
Any POD Basis vector needs to be an eigenvector of the POD operator R.

Proof.
We make use of the Lagrangian Method for constrained optimization problems and transform the resulting necessary condition into an eigenvalue problem for the POD operator R.

• Necessary Condition in Lagrange Context The Lagrange functional for the constraint optimization problem (2.6) reads:

$$L(\psi) = \left\langle |(y(t),\psi)_X|^2 \right\rangle_{t\in\Gamma} - \lambda(\|\psi\|_X^2 - 1)$$

A necessary condition for this functional to be minimal is of course that the functional derivative vanishes for all variations $\psi + \delta\mu \in X$, $\mu \in X, \delta \in \mathbb{R}$

$$\frac{d}{d\delta} L(\psi + \delta\mu)|_{\delta=0} = 0.$$

• Calculation of Derivative of L Observe

$$\begin{aligned}\frac{d}{d\delta} L(\psi+\delta\mu)|_{\delta=0} &= \frac{d}{d\delta} \langle (y(t),\psi+\delta\mu)_X\,(\psi+\delta\mu,y(t))_X \rangle_{t\in\Gamma} - \lambda\,(\psi+\delta\mu,\psi+\delta\mu)_X\,|_{\delta=0} \\ &= 2\left(\langle (y(t),\mu)_X\,(\psi,y(t))_X \rangle_{t\in\Gamma} - \lambda\,(\psi,\mu)_X\right) = 0.\end{aligned}$$

• Transformation to EVP Form If we now use commutativity of the average operator and the projection (represented by the inner product), we obtain:

$$\begin{aligned}\langle (y(t),\mu)_X\,(\psi,y(t))_X \rangle_{t\in\Gamma} - \lambda\,(\psi,\mu)_X &= \left\langle (y(t)\,(y(t),\psi)_X\,,\mu)_X \right\rangle_{t\in\Gamma} - \lambda\,(\psi,\mu)_X \\ &= \left(\langle y(t)\,(y(t),\psi)_X \rangle_{t\in\Gamma} - \lambda\psi,\mu\right)_X.\end{aligned} \tag{2.7}$$

Since $\mu \in X$ was arbitrary, we conclude

$$\langle y(t)\,(y(t),\psi)_X\rangle_{t\in\Gamma} = \lambda\psi,$$

which is precisely the eigenvalue problem for the operator R. □

2.2.2 Characterization of a POD Basis

We wish to establish a condition on $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ which is *equivalent* to saying that $\mathcal{B}^\ell$ denotes a POD Basis (of rank ℓ). For that purpose, we shall establish a sufficient condition for $\mathcal{B}^\ell$ being a POD Basis and comment on the equivalence. We may then use this new characterization to comment on the "quality" of the POD representation.

Sufficient Condition for a POD Basis Quite surprisingly, sufficient conditions cannot be established by verifying the second-order sufficient optimality conditions for the POD optimality problem (refer to Volkwein 2001*b*, p. 87 for details).

Hence, let us establish the actual characterization of a POD Basis by "calculating" that a carefully selected set of solutions to the first-order necessary condition actually solves the "POD Problem" of Definition 2.1.5. In particular, let us show:

Theorem 2.2.3 (Solution of Abstract POD Problem)
Let $\{\lambda_k\}_{k\in\mathbb{N}}$ be a (decreasingly) ordered set of eigenvalues and $\mathcal{B} = \{\lambda_k\}_{k\in\mathbb{N}}$ an orthonormal set of corresponding eigenvectors of the POD operator (defined in (2.5))

$$R : X \to \mathcal{V}, \quad R\psi = \langle y(t)\,(y(t),\psi)_X\rangle_{t\in\Gamma}\,,$$

such that $\mathcal{B}$ denotes a basis of $\mathcal{V}$.
Then $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ (i.e. an orthonormal set of eigenvectors of R corresponding to the ℓ first (largest) eigenvalues) denotes a POD Basis of rank ℓ.

Proof.
Suppose we are given $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ as defined in the assertion. Furthermore, let $\{\varphi_k\}_{k=1}^\ell \subset \mathcal{V}$ be an arbitrary ℓ-dimensional orthonormal set in $\mathcal{V}$.

• Idea The idea is to evaluate the objective of the POD Problem J on these two sets and compare the values:

$$J(\varphi_1,\ldots,\varphi_\ell) := -\left\langle \sum_{k=1}^{\ell} (y(t),\varphi_k)_X^2 \right\rangle_{t\in\Gamma} = -\sum_{k=1}^{\ell} \left\langle (y(t),\varphi_k)_X^2 \right\rangle_{t\in\Gamma} \overset{!}{\geq} J(\psi_1,\ldots,\psi_\ell). \tag{2.8}$$

• Reformulate Problem In order to do that, we shall *express φ_k in terms of ψ_k*. Since $\{\psi_k\}_{k\in\mathbb{N}}$ is an orthonormal basis, we may denote this as

$$\varphi_k = \sum_{j=1}^{\infty} (\varphi_k,\psi_j)_X\,\psi_j \quad \text{for } k = 1,\ldots,\ell \tag{2.9}$$

and investigate all additive terms in J. For $k = 1, \dots, \ell$, we find:

$$\begin{aligned}\left\langle (y(t), \varphi_k)_X^2 \right\rangle_{t\in\Gamma} &= \langle (y(t), \varphi_k)_X \, (y(t), \varphi_k)_X \rangle_{t\in\Gamma} = \left(\langle (y(t), \varphi_k)_X \, y(t) \rangle_{t\in\Gamma}, \varphi_k \right)_X \\ &= (R\varphi_k, \varphi_k)_X = \left(R \sum_{j=1}^{\infty} (\varphi_k, \psi_j)_X \, \psi_j, \varphi_k \right)_X \\ &\overset{EV}{=} \left(\sum_{j=1}^{\infty} (\varphi_k, \psi_j)_X \, \lambda_{\mathbf{j}} \psi_j, \varphi_k \right)_X \\ &= \sum_{j=1}^{\infty} \lambda_j \, (\psi_j, \varphi_k)_X^2 \, . \end{aligned} \tag{2.10}$$

• Estimation by a Truncated Sum Let us expand the latter expression in order to estimate it from below by sums of ℓ summands. First, we observe (since $\{\varphi_k\}_{k=1}^{\ell}$ is an orthonormal set):

$$1 = (\varphi_k, \varphi_k)_X = \left(\sum_{j=1}^{\infty} (\varphi_k, \psi_j)_X \, \psi_j, \varphi_k \right)_X = \sum_{j=1}^{\infty} (\psi_j, \varphi_k)_X^2 \geq \sum_{j=1}^{\ell} (\psi_j, \varphi_k)_X^2 \, . \tag{2.11}$$

Thus, adding $0 = \lambda_\ell - 1\lambda_\ell$ to (2.10), we obtain

$$\begin{aligned}\left\langle (y(t), \varphi_k)_X^2 \right\rangle_{t\in\Gamma} &= \sum_{j=1}^{\infty} \lambda_j \, (\psi_j, \varphi_k)_X^2 + \lambda_\ell - \lambda_\ell \underbrace{\sum_{j=1}^{\infty} (\psi_j, \varphi_k)_X^2}_{=1 \, (2.11)} \\ &= \lambda_\ell + \sum_{j=1}^{\ell} \lambda_j \, (\psi_j, \varphi_k)_X^2 - \lambda_\ell \sum_{j=1}^{\ell} (\psi_j, \varphi_k)_X^2 \\ &\quad - \left(\lambda_\ell \sum_{j=\ell+1}^{\infty} (\psi_j, \varphi_k)_X^2 - \sum_{j=\ell+1}^{\infty} \lambda_j \, (\psi_j, \varphi_k)_X^2 \right) \\ &\leq \lambda_\ell + \sum_{j=1}^{\ell} \lambda_j \, (\psi_j, \varphi_k)_X^2 - \lambda_\ell \sum_{j=1}^{\ell} (\psi_j, \varphi_k)_X^2 \, , \end{aligned} \tag{2.12}$$

since the term in brackets is non-negative as $\lambda_\ell \geq \lambda_j$ for all $j \geq \ell + 1$.

• Calculate Functional Value Observe that for the special case $\varphi_k := \psi_k$, equation (2.10) yields

$$\left\langle (y(t), \psi_k)_X^2 \right\rangle_{t\in\Gamma} = \sum_{j=1}^{\infty} \lambda_j \underbrace{(\psi_j, \psi_k)_X}_{=\delta_{jk}}{}^2 = \lambda_k \quad \text{for all } k \in \mathbb{N}. \tag{2.13}$$

Thus, we infer for the functional value applied to $\mathcal{B}^\ell$:

$$J(\psi_1, \dots, \psi_\ell) = -\sum_{j=1}^{\ell} \left\langle (y(t), \psi_j)_X^2 \right\rangle_{t\in\Gamma} = -\sum_{j=1}^{\ell} \lambda_j. \tag{2.14}$$

• Show assertion By combining (2.8) and (2.12) (and minding the minus sign), we obtain

$$\begin{aligned} J(\varphi_1,\dots,\varphi_\ell) &= -\sum_{k=1}^{\ell}\left\langle (y(t),\varphi_k)_X^2 \right\rangle_{t\in\Gamma} \\ &\geq -\sum_{k=1}^{\ell}\Big(\lambda_\ell + \sum_{j=1}^{\ell}\lambda_j\,(\psi_j,\varphi_k)_X^2 - \lambda_\ell\sum_{j=1}^{\ell}(\psi_j,\varphi_k)_X^2\Big) \\ &= -\ell\lambda_\ell - \sum_{j=1}^{\ell}\Big(\lambda_j\sum_{k=1}^{\ell}(\psi_j,\varphi_k)_X^2 - \lambda_\ell\sum_{k=1}^{\ell}(\psi_j,\varphi_k)_X^2\Big) \\ &= -\sum_{j=1}^{\ell}\Big(\lambda_\ell + (\lambda_j-\lambda_\ell)\sum_{k=1}^{\ell}(\psi_j,\varphi_k)_X^2\Big) \\ &\geq -\sum_{j=1}^{\ell}\Big(\lambda_\ell + (\lambda_j-\lambda_\ell)\Big) \\ &= -\sum_{j=1}^{\ell}\lambda_j \\ &= J(\psi_1,\dots,\psi_\ell), \end{aligned} \tag{2.15}$$

where the last estimation is due to (2.11) and the last step is given by (2.14); completing the proof. □

Equivalence Issues We are close to saying that the eigenvectors of the operator R *precisely* characterize a POD Basis. This motivates of the following corollary.

Corollary 2.2.4 (Equivalence of Characterization)
$\mathcal{B}^\ell = \{\psi_k\}_{k=1}^{\ell}$ denotes a POD Basis for $\mathcal{V}$ *if and only if* $\mathcal{B} = \{\psi_k\}_{k\in\mathbb{N}}$ denotes a set of eigenvectors of R (ordered by the magnitude of the corresponding eigenvalues and forming a Basis for $\mathcal{V}$).

Proof.
We have justified the sufficiency of the "eigenvector criterion" in Theorem 2.2.3 already. Thus, we only have to take care of the necessity. Since the respective proof becomes quite technical and the result is not central to the theory in this context, we only give a sketch of the proof. (A more detailed proof may be found in Volkwein 2001*b*, Section 2 for a complex Hilbert space X and a finite ensemble or in Volkwein 2006, Theorem 1.1 for $X = \mathbb{R}^n$, where the proof for this general case works perfectly analogous to the latter one.)
In a similar fashion to the proof of Proposition 2.2.2, we apply the Lagrangian method to an "optimality system" for the case of ℓ POD modes and derive an eigenvalue problem for the operator R.
Some care has to be taken when setting up the Lagrangian functional: We have to include all (pair-wise) orthogonality conditions, i.e. ℓ^2 terms of the sort $\delta_{ik} - (\psi_i, j)_X$, $i, j = 1,\dots,\ell$.
We then transform the resulting necessary condition into an eigenvalue problem and proceed by induction over $k = 1,\dots,\ell$ in order to show that every POD mode has to fulfill the eigenvector criterion of Theorem 2.2.3. (The start of the induction basically is given by Proposition 2.2.2.)

Then, every POD mode is precisely characterized by the eigenvector characterization proposed. □

2.2.3 Error of a POD Basis Representation

Let us present an error estimate for the approximation of a given ensemble $\mathcal{V}$ by a POD Basis of rank ℓ. Note that this *only* measures the error of representing the members of $\mathcal{V}$ by the POD Basis and is *only an ingredient* for an error estimate for a Reduced-order Model or such.

Best Approximation – Full Basis of Eigenvectors Let us motivate why a *complete* basis of R-eigenvectors for X is handy for formulating an expression for the error of representation.
In particular, we are interested in the error $\left\|g - g^\ell\right\|_X$ of the best approximation $g^\ell \in \mathcal{V}^\ell$ of an element $g \in \mathcal{V}$. By means of the POD projection, we may represent any $g^\ell \in \mathcal{V}^\ell$ in Fourier form and by means of the *complete basis of eigenvectors*, we may setup a similar statement for each ensemble member $g \in \mathcal{V}$:

$$g^\ell = P^\ell g = \sum_{k=1}^{\ell} (g, \psi_k)_X \, \psi_k \quad \text{and} \quad g = \sum_{k=1}^{\infty} (g, \psi_k)_X \, \psi_k.$$

By means of these equations, we do not only clearly see that g^ℓ is an approximation for g, yet also may easily find an expression for the term $g - g^\ell$ which actually is of interest.

Error in Highest Contribution Note that we have "implicitly" derived the maximal value of the "highest contribution" version of the POD Problem in (2.14) already:

$$\operatorname{argmax}_{\mathcal{B}^\ell} \sum_{j=1}^{\ell} \left\langle (y(t), \psi_j)_X^2 \right\rangle_{t\in\Gamma} = \sum_{j=1}^{\ell} \lambda_j.$$

Error in Best Approximation In context of Model Reduction, we actually are interested in the error of representation, i.e. the minimal value of the "best approximation" version of the POD Problem. Since according to Proposition 2.1.6 the extremal values of the two problem statements differ, let us establish the following proposition:

Proposition 2.2.5 (POD Representation Error)
Let $\{\lambda_k\}_{k\in\mathbb{N}}$ be an ordered set of eigenvalues of the operator R and $\{\psi_k\}_{k=1}^{\ell}$ a POD Basis, i.e. an orthonormal eigenvectors of R corresponding to the ℓ first (largest) eigenvalues.
By definition of the POD Problem the error of the POD approximation is given by the minimal value of the functional J, for which holds

$$\operatorname{argmin} J = \left\langle \left\| y(t) - \sum_{k=1}^{\ell} (y(t), \psi_k)_X \, \psi_k \right\|_X^2 \right\rangle_{t\in\Gamma} = \sum_{k=\ell+1}^{\infty} \left\langle (y(t), \psi_k)_X^2 \right\rangle_{t\in\Gamma} = \sum_{k=\ell+1}^{\infty} \lambda_k. \tag{2.16}$$

Proof.
Let us proceed in two steps, essentially using previous work.

• Calculation of $\|g\|_X^2$ Observe that since $\{\psi_k\}_{k\in\mathbb{N}}$ is an orthonormal basis of $\mathcal{V}$, we may write

$$g = \sum_{k=1}^{\infty} (g, \psi_k)_X \psi_k \quad \text{for all } g \in \mathcal{V}.$$

It follows for the norm of each $g \in \mathcal{V}$:

$$\|g\|_X^2 = (g,g)_X = \sum_{k=1}^{\infty}\sum_{j=1}^{\infty} (g,\psi_k)_X\,(g,\psi_j)_X \underbrace{(\psi_k,\psi_j)_X}_{=\delta_{kj}} = \sum_{k=1}^{\infty} (g,\psi_k)_X^2\,. \tag{2.17}$$

• Showing the Assertion In the following calculation, the first step is due to the proof of Proposition 2.1.6, in the second step we use (2.17) and the very last step is due to (2.13). In particular, we obtain (with $y(t) \in \mathcal{V}$)

$$\begin{aligned}
\left\langle \left\| y(t) - \sum_{k=1}^{\ell} (y(t),\psi_k)_X\,\psi_k \right\|_X^2 \right\rangle_{t\in\Gamma} &= \left\langle \|y(t)\|_X^2 - \sum_{k=1}^{\ell} (y(t),\psi_k)_X{}^2 \right\rangle_{t\in\Gamma} \\
&= \left\langle \sum_{k=1}^{\infty} (y(t),\psi_k)_X^2 - \sum_{k=1}^{\ell} (y(t),\psi_k)_X^2 \right\rangle_{t\in\Gamma} \\
&= \left\langle \sum_{k=\ell+1}^{\infty} (y(t),\psi_k)_X^2 \right\rangle_{t\in\Gamma} \\
&= \sum_{k=\ell+1}^{\infty} \left\langle (y(t),\psi_k)_X^2 \right\rangle_{t\in\Gamma} = \sum_{k=\ell+1}^{\infty} \lambda_k.
\end{aligned}$$

□

2.2.4 Existence of a POD Basis

We shall now show that there actually exists a set $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^{\ell}$ that fulfills the characterization of a POD Basis in Theorem 2.2.3 (and Corollary 2.2.4).

Procedure We establish a basis $\mathcal{B} = \{\psi_k\}_{k\in\mathbb{N}}$ for $\mathcal{V}^\ell$, which consists of eigenvectors of the operator R. For this purpose, we shall make use of the Hilbert Schmidt Theorem, which essentially leaves us with showing that R is a *self-adjoint, compact* operator. This fact we shall justify by decomposing R in such a way that it becomes obvious that R is self-adjoint.

Preparation Let us make use of the Hilbert Schmidt Theorem (proved in Reed and Simon 1980, Theorem VI.16 for example) in order to transform our objective to showing that R is a self-adjoint, compact operator:

Lemma 2.2.6 (Hilbert-Schmidt Theorem)
Let $R : X \to X$ be a self-adjoint, compact operator on a separable Hilbert space X. Then, there exists a complete orthonormal basis $\{\psi_k\}_{k\in\mathbb{N}}$ for X so that $R\psi_k = \lambda_k\psi_k$ (for $k \in \mathbb{N}$) and $\lambda_k \to 0$ as $k \to \infty$.

Decomposition of R We define a "decomposition operator" such that later on, we may easily show that R is self-adjoint and non-negative. (Actually we benefit from this ansatz even more by giving an alternative characterization of a POD Basis; refer to Subsection 2.2.5. Even further insides on the respective operators shall be given in Chapter A.) In order to simply the presentation, we restrict ourselves to "trivial" weight functions.

Proposition 2.2.7 (Decomposition of R)
Let $(\Gamma, y, \mathcal{V}_P)$ be a parametrized ensemble and let $\omega_0 \in L^2(\Gamma)$ be a weighting function of constant value 1.
Then, the bounded *decomposition operator* $\mathcal{Y} : L^2(\Gamma) \to X$, defined by

$$\mathcal{Y}v := \left\langle \omega_0, (v,y)_{L^2(\Gamma)} \right\rangle_{\mathcal{V}_P} = \left(\omega_0, (v,y)_{L^2(\Gamma)} \right)_{L^2(\Gamma)} = (v,y)_{L^2(\Gamma)} \quad \text{for } v \in L^2(\Gamma)$$

and its adjoint $\mathcal{Y}^* : X \to L^2(\Gamma)$, which is given by

$$(\mathcal{Y}^* z)(t) := (y(t), z)_X \quad \text{for } z \in X,$$

decompose the operator R such that there holds

$$R = \mathcal{Y}\mathcal{Y}^*.$$

Proof.
Clearly, the operator $\mathcal{Y}$ is bounded since the inner product in $L^2(\Gamma)$ is bounded. (The transformation of $\mathcal{Y}$ is due to Definition 2.1.3 of the average operator and the choice of weight function ω_0.) Hence, let us focus on the other assertions.

• Adjoint Result Note that $\mathcal{Y}^*$ is the adjoint of $\mathcal{Y}$ since for all $v \in L^2(\Gamma)$ and $z \in X$ there holds

$$(\mathcal{Y}v, z)_X = \left((v,y)_{L^2(\Gamma)}, z \right)_X = \left(v, (y,z)_X \right)_{L^2(\Gamma)} = (v, \mathcal{Y}^* z)_{L^2(\Gamma)}.$$

• Decomposition of R We simply have to compute $\mathcal{Y}\mathcal{Y}^*$. Using the simplified notation of the average operator of Definition 2.1.4 and the trivial choice for ω_0, we may show the assertion:

$$\begin{aligned} \mathcal{Y}\mathcal{Y}^* z = (Y^* z, y)_{L^2(\Gamma)} &= \int_\Gamma (Y^* z)(t) y(t) \, \mathrm{d}t = \int_\Gamma (y(t), z)_X \, y(t) \, \mathrm{d}t \\ &= \left\langle y(t) \, (y(t), z)_X \right\rangle_{t \in \Gamma} = Rz \quad \text{for all} \quad z \in X. \end{aligned}$$

□

Existence of a POD Basis We may now show the existence of a complete orthonormal basis of X which consists of eigenvectors of the operator R. Then, Theorem 2.2.3 teaches us how to construct a POD Basis from that basis.

Proposition 2.2.8 (Properties and Spectral Decomposition of R)
The POD operator R is linear, bounded, non-negative, self-adjoint and compact.
There exists a *complete orthonormal basis* of X consisting of eigenvectors $\{\psi_i^\infty\}_{i \in \mathbb{N}}$ of R and a corresponding sequence $\{\lambda_i^\infty\}_{i \in \mathbb{N}}$ of non-negative real eigenvalues of R.
The spectrum of R is a pure point spectrum, except for possibly 0. Each nonzero eigenvalue of R has finite multiplicity and 0 is the only possible accumulation point of the spectrum of R.

Proof.
Clearly, R is linear and bounded since $\mathcal{Y}$ is bounded. For the other assertions, we find:

• Properties of R Since every Hilbert space is reflexive, we by virtue of Kato 1980, V-(2.1) have $\mathcal{Y}^{**} = \mathcal{Y}$. Then, due to the decomposition established above, R is *self-adjoint*

$$R^* = (\mathcal{Y}\mathcal{Y}^*)^* = \mathcal{Y}^{**}\mathcal{Y}^* = \mathcal{Y}\mathcal{Y}^* = R$$

and *non-negative*:

$$(Rv, v)_X = (\mathcal{Y}\mathcal{Y}^* v, v)_X = (\mathcal{Y}^* v, \mathcal{Y}^* v)_X = \|\mathcal{Y}^* v\|_X^2 \geq 0 \quad \text{for all } v \in X.$$

The compactness of R may be established as follows: Since $g \in C(\Gamma; V)$ (see Definition 2.1.1) holds, the *Kolmogorov compactness criterion* in $L^2(\Gamma)$ implies that $Y^* : X \to L^2(\Gamma)$ is compact. The boundedness of $\mathcal{Y}$ then implies that R is a compact operator as well (refer to Kato 1980, Theorem III-4.8).

• Eigenvector Basis From the Hilbert-Schmidt theorem (refer to Lemma 2.2.6) and non-negativeness of R it follows that there exists a *complete, orthonormal basis* $\{\psi_i^\infty\}_{i\in\mathbb{N}}$ for X and a sequence $\{\lambda_i^\infty\}_{i\in\mathbb{N}}$ of non-negative real numbers so that

$$R_\infty \psi_i^\infty = \lambda_i^\infty \psi_i^\infty, \quad \lambda_1^\infty \geq \lambda_2^\infty \geq \cdots \geq 0 \quad \text{and} \quad \lambda_i^\infty \to 0 \text{ as } i \to \infty. \tag{2.18}$$

• Spectrum Considerations The assertions are justified in Kato 1980, Theorem III-6.26 on page 185. □

Non-Uniqueness of a POD Basis Let us note that due to the equivalent characterization of the POD Basis as eigenvectors of R we conclude that the POD Basis may not be uniquely determined:

Remark 2.2.9 (Free Choices in Eigenvector Basis)
Let us note that in case the operator R has got degenerated eigenvalues, we may freely choose an orthonormal basis for the respective "eigen spaces" and hence, the POD Basis cannot be unique in general.

2.2.5 Alternative Characterization of a POD Basis

In this subsection, we wish to show a different way of calculating a POD Basis. (This shall become handy for practical applications. Yet at this point, we only give the abstract formulation which we aim to interpret further in Chapter 3.)

Alternative EVP to Characterize a POD Basis Since a POD Basis essentially is given by eigenvectors of the operator R, the following proposition "implicitly" claims that we may compute a POD Basis by means of a so-called *correlation operator* K (for more details on the issue of correlation refer to Chapter A).

Proposition 2.2.10 (Alternative POD Operator)
Define the *correlation operator* $K : L^2(\Gamma) \to L^2(\Gamma)$ by

$$K := \mathcal{Y}^*\mathcal{Y}, \qquad (Kv)(t) = \int_\Gamma (y(t), y(s))_X \, v(s) \, \mathrm{d}s \quad \text{for } v \in L^2(\Gamma). \tag{2.19}$$

Then, the operator K admits the same properties as R does in Proposition 2.2.8. Moreover, except for possibly 0, K and R possess the same eigenvalues which are positive with identical multiplicities. Furthermore, an eigenvector ψ_k of R and an eigenvector

v_k of K may be converted into each other by (for all $t \in \Gamma$):

$$v_k(t) = \frac{1}{\sqrt{\lambda_k}}(\mathcal{Y}^*\psi_k)(t) = \frac{1}{\sqrt{\lambda_k}}\,(y(t), \psi_k)_X \quad \text{and} \quad \psi_k = \frac{1}{\sqrt{\lambda_k}}\mathcal{Y}v_k. \tag{2.20}$$

Proof.
Let us first investigate the newly defined operator K and then look at the relation to the operator R.

• Statement of K We may transform

$$(Kv)(t) = (\mathcal{Y}^*\mathcal{Y})(t) = \left(y(t), \int_\Gamma y(s)v(s)\,\mathrm{d}s\right)_X \quad \text{for } v \in L^2(\Gamma)$$

by interchanging the integration with the inner product in X into

$$(Kv)(t) = \int_\Gamma (y(t), y(s))_X\, v(s)\,\mathrm{d}s \quad \text{for } v \in L^2(\Gamma).$$

• Properties of K K is linear, bounded, self-adjoint and non-negative, which might be shown analogously to the proof of Proposition 2.2.8 (which claims the operator R to have these properties).

• Results on Eigenvalue Sets of K and R Refer to Kunisch and Volkwein 2006, Proposition 2.1 for example.

• Conversion Formula for Eigenvectors Let us omit the index k of the eigenvectors in order to simplify the notation. For v an eigenvector of $K = \mathcal{Y}^*\mathcal{Y}$ and ψ an eigenvector of $R = \mathcal{Y}\mathcal{Y}^*$, it follows rather easily that $\mathcal{Y}v$ and $\mathcal{Y}^*\psi$ are eigenvectors for R and K, respectively:

$$\begin{aligned} R(\mathcal{Y}v) = \mathcal{Y}(\mathcal{Y}^*\mathcal{Y})v = \mathcal{Y}Kv = \lambda(\mathcal{Y}v) \\ \text{and} \quad K(\mathcal{Y}^*\psi) = \mathcal{Y}^*(\mathcal{Y}\mathcal{Y}^*)\psi = \mathcal{Y}^*R\psi = \lambda(\mathcal{Y}^*\psi) \end{aligned}$$

In order to ensure the *"unity"* of the respective eigenvectors we introduce a normalization factor $\frac{1}{\sqrt{\lambda}}$, since for the norm of $\psi := \mathcal{Y}v$ we obtain

$$\|\psi\|_X^2 = (\psi, \psi)_X = (\mathcal{Y}v, \mathcal{Y}v)_X = (v, \mathcal{Y}^*\mathcal{Y}v)_X = (v, Kv)_X = \lambda\,(v, v)_X = \lambda.$$

□

Generalized SVD Decomposition Note that the relation (2.20) is "formally" equivalent to the characterization of "singular vectors" in (1.1). These vectors form an *SVD* of a matrix Y (refer to Theorem 1.1.2). Hence, we find that Proposition 2.2.10 presents a *generalized form* of SVD of the operator $\mathcal{Y}$, which motivates the following

Remark 2.2.11 (Calculation of a POD Basis by SVD)
In a discrete context, it is possible to calculate a POD Basis by an SVD of a suitable matrix Y (refer to Subsection 3.1.2, in particular Theorem 3.1.5).

2.3 Asymptotic Behaviour of the POD Error

In practical applications, we of course only deal with a finite number of ensemble members, i.e. with a *finite* ensemble grid Γ_n. It shall turn out that the corresponding POD error estimate depends on the actual choice of Γ_n.
In this section, we hence wish to find an error estimate which is independent of the ensemble grid. In particular, we wish the "finite problem" to converge to a "continuous" one. (A "spin-off effect" of this objective will be to better understand the optimality properties of the POD Method; refer to Subsection 4.3.1.) Most of the theory may for example be found in Kunisch and Volkwein 2002, Section 3.2.

Procedure We restrict the abstract POD Problem by setting $\Gamma_n := \{t_j\}_{j=0}^n \in \mathbb{N}$ as well as $\Gamma_\infty := [0,T]$ (where we indicate their later usage by the choice of notation). Since these two cases differ only slightly, we workout the respective problems and corresponding solution operators in a "side-by-side" fashion by refining the general results from Section 2.2.
We then show the convergence of the finite problem to the continuous one and may finally estimate that the error for any finite set Γ_n is bounded by the error for Γ_∞. In this sense, the error for any finite set Γ_n becomes "independent" of Γ_n.
We conclude by commenting on the optimality properties of the POD Method in context of Evolution Systems.

2.3.1 Treatment of Problems and Solutions

Let us introduce all the ingredients of a POD Problem for the respective cases.

The Ensemble For the particular choices $\Gamma_n := \{t_j\}_{j=0}^n \subset \mathbb{N}$ and $\Gamma_\infty := [0,T]$, we obtain the ensemble sets

$$\mathcal{V}_P^n = \{y_1, \dots, y_n\} \subset X \quad \text{and} \quad \mathcal{V}_P^\infty = \{y(t) | t \in [0,T]\}$$

and in turn define the ensemble spaces $\mathcal{V}^\infty := \operatorname{span}(\mathcal{V}_P^\infty)$ as well as $\mathcal{V}^n := \operatorname{span}(\mathcal{V}_P^n)$ with $\dim \mathcal{V}^n = d \leq n$ (since the ensemble members might be linearly dependent in X).
Furthermore, we introduce respective parameterizations. (Note the additional requirement on the derivative of the parametrization of $\mathcal{V}_P^\infty$ which is due to "convergence reasons".)

$$y^n : \{t_j\}_{j=0}^n \to \mathcal{V}_P^n \qquad \text{and} \qquad y^\infty \in C([0,T]; \mathcal{V}_P^\infty) \quad \text{with} \quad y_t^\infty \in L^2([0,T]; X). \tag{2.21}$$

Projection Operator We still operate in the same Hilbert space and still try to find an $(\cdot,\cdot)_X$-orthonormal basis $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ to solve the POD Problem. Thus, the projection remains unchanged for both the cases and (still) reads:

$$P^\ell v = \sum_{k=1}^{\ell} (v, \psi_k)_X \, \psi_k \quad \text{for all } v \in X. \tag{2.22}$$

Average Operator Due to the different choices for Γ, the average operators do differ. For convergence reasons (see below), we weight the average in the discrete case by $\alpha_j \in \mathbb{R}$ for $j = 0, \dots, n$, whose choice is discussed in Remark 2.3.6. (For now, think of weights of a "quadrature formula" such as the trapezoidal rule.)
In terms of Definition 2.1.3 of the average operator, we choose weighting functions:

$$\omega_n : \{t_j\}_{j=0}^n \to \mathbb{R}, \quad \omega_n(t_j) := \frac{1}{n}\alpha_j, \quad j = 0, \dots, n \qquad \text{and} \qquad \omega_\infty \in L^2([0,T]), \quad \omega_\infty(t) \equiv \frac{1}{T}.$$

Then, (for the simplified notation of Definition 2.1.4 and a suitable functional F) we arrive at

$$\langle F(y^n(t))\rangle_{t\in\Gamma_n} = \frac{1}{n}\sum_{j=1}^{n}\alpha_j F(y_j) \quad \text{and} \quad \langle F(y^\infty(t))\rangle_{t\in\Gamma_\infty} = \frac{1}{T}\int_0^T F(y^\infty(t))\,\mathrm{d}t. \tag{2.23}$$

Problem Statement Let us now apply the choices above to the general Definition 2.1.5 for the *best approximation* version. (Note that we may omit the constant factors $1/n$ and $1/T$ and do not explicitly mention the definition of F in the definitions of average operators.) In particular, we arrive at

Problem 2.3.1 (Finite and Infinite POD Problem)
Find an *orthonormal* basis $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ that fulfills in the finite case

$$(\Gamma_n, y^n, \mathcal{V}_P^n, P^\ell, \langle\cdot\rangle_{t\in\Gamma_n}), \qquad \min_{\mathcal{B}^\ell} J_n := \sum_{j=1}^{n}\alpha_j \left\| y_j - \sum_{k=1}^{\ell}(y_j,\psi_k)_X\,\psi_k \right\|_X^2 \tag{2.24}$$

and in the infinite case

$$(\Gamma_\infty, y^\infty, \mathcal{V}_P^\infty, P^\ell, \langle\cdot\rangle_{t\in\Gamma_\infty}), \qquad \min_{\mathcal{B}^\ell} J_\infty := \int_0^T \left\| y(t) - \sum_{i=1}^{\ell}(y(t),\psi_i)_W\,\psi_i \right\|_X^2 \mathrm{d}t. \tag{2.25}$$

Classical POD Solutions Let us simply derive the forms of the POD Operator of Theorem 2.2.3 for the respective choices of average operators. We therewith obtain:

Proposition 2.3.2 (Solution of (Asymptotic) POD Problem)
From the POD operator $R = \langle (v, y(t))_X\, y(t)\rangle_{t\in\Gamma}$ derive the operators

$$R_n : X \to \mathcal{V}^n, \qquad R_n v := \left\langle (v, y^n(t))_X\, y^n(t)\right\rangle_{t\in\Gamma_n} = \sum_{j=1}^{n}\alpha_j\,(v,y_j)_X\,y_j \quad \text{for } v\in X,$$

$$R_\infty : X \to X, \qquad R_\infty z := \left\langle (z, y^\infty(t))_X\, y^\infty(t)\right\rangle_{t\in\Gamma_\infty} = \int_0^T (z, y^\infty(t))_X\, y^\infty(t)\,\mathrm{d}t \quad \text{for } z\in X.$$

Denote the eigenfunctions and corresponding eigenvalues of the operator R_n by

$$\{\psi_k\}_{k=1}^\infty \text{ and } \{\lambda_k\}_{k=1}^\infty \text{ with } \lambda_1 \geq \lambda_2 \geq \cdots \geq 0.$$

Then, the first ℓ eigenvectors $(\psi_1,\ldots,\psi_\ell)$ are a POD Basis of rank ℓ in the sense of (2.24).
Similarly, a solution to (2.25) is given by the eigenvectors $\{\psi^\infty\}_{i=1}^\ell$ of R_∞ corresponding to the ℓ largest eigenvalues $\lambda_1^\infty \geq \cdots \geq \lambda_\ell^\infty$.

Proof.
The image space of R_n is well-defined since R_n yields a linear combination of $y_i \in \mathcal{V}_P^n$. Since we have derived R_n and R_∞ from the "general" POD operator R, may choose the respective eigenvalues and eigenvectors due to Proposition 2.2.8. Then, we may simply make use of the characterization of a POD Basis in Theorem 2.2.3 for this more concrete case in order to establish the claim about the POD Basis.
(An explicit derivation of the discrete case for $\alpha_j \equiv 1$, $j = 1,\ldots,n$, may be found in Volkwein 2001*b*, Theorem 3 for example.) □

Restriction of R_n Would Suffice Let us remark that the EVP for R_n only is of infinite dimension in the theoretical context, where we wish to establish a *complete* basis of eigenvectors for X in order to obtain a basis for $X \setminus \mathcal{V}^n$, too (refer to Subsection 2.2.3).

Remark 2.3.3 (EVP for R_n Essentially is of Finite Dimension)
Looking at the EVP for R_n more carefully, we infer that all eigenvectors ψ to an eigenvalue $\lambda \neq 0$ have to lie in $\mathcal{V}^n$, which is of dimension $d < \infty$ (since for $0 \neq \lambda \in \mathbb{R}$ and ψ an eigenvector of R_n there holds: $\mathcal{V}^n \ni R_n\psi = \lambda\psi$ implies $\psi \in \mathcal{V}^n$).
Hence, to determine a POD Basis it would suffice to consider the restricted operator: $R_n\,|_{\mathcal{V}^n} \colon \mathcal{V}^n \to \mathcal{V}^n$.

Error Estimate for POD Representation Let us now look at the resulting error of the approximation, in particular at the *drawback* of the dependence on the ensemble grid $\Gamma_n = \{t_j\}_{j=0}^n$ in the discrete case – which essentially is the motivation of this section.
Recall that the error of the POD representation by definition of the "best approximation form" of a POD Problem is given by the minimal value of the respective POD functionals. In particular, for the respective choices in the POD Problems and the dimension $d(n)$ of $\mathcal{V}^n$, we directly infer from Proposition 2.2.5:

Corollary 2.3.4 (Error Estimate)
For the solutions to the POD Problems in Problem 2.3.1, we obtain for the minimal values of the respective functionals J_n and J_∞:

$$\operatorname{argmin} J_n = \sum_{k=\ell+1}^{d(n)} \lambda_k^n \qquad \text{and} \quad \operatorname{argmin} J_\infty = \sum_{k=\ell+1}^{\infty} \lambda_k^\infty, \tag{2.26}$$

where the eigenvalues λ_k^n depend on the actual choice of the ensemble set $\Gamma_n = \{t_j\}_{j=0}^n$.

2.3.2 Convergence of POD Solutions

Let us now investigate the dependence of the error estimate on the finite ensemble grid $\Gamma_n = \{t_j\}_{j=0}^n$, found in Corollary 2.3.4.

Mathematical Procedure In mathematical terms, the convergence of the POD Problems can be realized by investigating the problem (2.24) for the ensemble grid size Δt converging to 0 (refer to Definition 2.1.1).
The convergence of the problems as well as of the operators R_n to R_∞ will be ensured by the choice of the weights α_j. It remains to show that this implies that the actual POD solutions (the respective eigenvectors) converge as well.

Ensuring Convergence of Problems We may easily justify that (2.25) is appropriate as a "limit problem". Moreover, we note that the issue of convergence actually presents the only constraint on our choice of weights in the average operator.

Proposition 2.3.5 (Convergence of Problems)
For the ensemble grid size Δt in Definition 2.1.1 approaching zero, the problem (2.24) approaches problem (2.25).

Proof.
The functional J_α is the trapezoidal approximation of J_∞. So the convergence follows from basic numerical mathematics. □

Remark 2.3.6 (Choice of α_j)
In terms of the analysis, the choice of the weights α_j in the definition of the average operator (2.23) is arbitrary as long as convergence to the operator of the continuous problem can be achieved.

Convergence of Operators Note that $R_n\varphi$ is the *trapezoidal approximation* to the "integral" $R_\infty\varphi$. Hence, we obtain the following proposition, whose prerequisite is assumed in (2.21).

Proposition 2.3.7 (Convergence of operator R_n)
Let $y^\infty \in L^2([0,T];X)$. Then, we obtain

$$\lim_{\Delta t\to 0} \|R_n - R_\infty\|_{\mathcal{L}(X)} = 0. \tag{2.27}$$

Convergence of POD Solutions Recall that we denote by $\{\lambda_i^n\}_{i=1}^{d(n)}$ the positive eigenvalues of R^n with associated eigenfunctions $\{\psi_i^n\}_{i=1}^{d(n)}$. Similarly, $\{\lambda_i^\infty\}_{i\in\mathbb{N}}$ denotes the positive eigenvalues of R_∞ with associated eigenfunctions $\{\psi_i^\infty\}_{i\in N}$. In each case the eigenvalues are considered according to their multiplicity.

Proposition 2.3.8 (Convergence of Eigenvalue/Eigenvector)
Choose and fix ℓ such that $\lambda_\ell^\infty \neq \lambda_{\ell+1}^\infty$. Then, we obtain for the eigenvalues

$$\lim_{\Delta t\to 0} \lambda_i^n = \lambda_i^\infty \quad \text{for } i = 1,\dots,\ell \tag{2.28}$$

as well as for the eigenvectors

$$\lim_{\Delta t\to 0} \psi_i^n = \psi_i^\infty \quad \text{for } i = 1,\dots,\ell. \tag{2.29}$$

Proof.
The result follows due to (2.27) and by virtue of spectral analysis of compact operators (refer for example to Kato 1980, pp. 212–214). □

2.3.3 Treatment of Error Estimation

We are now able to give an *asymptotic estimate* for the POD error estimate (2.26) derived above. In particular, we investigate $\sum_{i=\ell+1}^{d(n)} \lambda_i$ as Δt tends to zero, i.e., $n \to \infty$.
Moreover, in the analysis of Reduced-order Modeling (refer to Section 4.2), we shall need an estimation of the projection of the initial value as well. Hence, let us provide a corresponding result at this stage, too.

Proposition 2.3.9 (Asymptotic Error Estimate)
If $\sum_{i=\ell+1}^{\infty} \lambda_i^\infty \neq 0$, there exists a $\overline{\Delta t} > 0$ such that for the error in the *POD approximation*

$$\sum_{i=\ell+1}^{d(n)} \lambda_i^n \leq 2 \sum_{i=\ell+1}^{\infty} \lambda_i^\infty \quad \text{for all } \Delta t \leq \overline{\Delta t}$$

and for the error in the *projection of the initial value*

$$\sum_{i=\ell+1}^{d(n)} |(\psi_i^n, y_0^n)_X|^2 \leq 2 \sum_{i=\ell+1}^{\infty} |(\psi_i^\infty, y_0^\infty)_X|^2 \quad \text{for all } \Delta t \leq \overline{\Delta t},$$

provided that $\sum_{i=\ell+1}^{\infty} \left|(y_0, \psi_i^\infty)_X\right|^2 \neq 0$.

Proof.
In order to establish the assertion on error of the POD approximation, we show that

$$\sum_{i=1}^{d(n)} \lambda_i^n \to \sum_{i=1}^{\infty} \lambda_i^\infty \quad \text{as } \Delta t \to 0, \tag{2.30}$$

which together with (2.28) implies the assertion. In order to do this we, roughly speaking, transform the problem into a convergence problem of a sum to an integral expression.

• Transformation Sum-Integral Expression By "formally" choosing $\ell := 0$ in the statements of the POD Problem 2.3.1, Corollary 2.3.4 yields (for every $n \in \mathbb{N}$, omitting the dependence of α_j on n):

$$\sum_{j=0}^{n} \alpha_j \left\|y^n(t_j)\right\|_X^2 = \sum_{i=1}^{d(n)} \lambda_i^n \quad \text{and} \quad \int_0^T \left\|y^\infty(t)\right\|_X^2 \, \mathrm{d}t = \sum_{i=1}^{\infty} \lambda_i^\infty. \tag{2.31}$$

• Showing Assertion Using this fact, we may transform (2.30) into

$$\sum_{j=0}^{n} \alpha_j \left\|y^n(t_j)\right\|_X^2 \to \int_0^T \left\|y^\infty(t)\right\|_X^2 \, \mathrm{d}t \quad \text{as } \Delta t \to 0, \tag{2.32}$$

which is true since we have assumed $y^\infty \in C([0,T]; \mathcal{V}_P^\infty)$.

• Initial Value Projection Error For a proof refer to Kunisch and Volkwein 2002, (3.15), p. 500. □

Chapter 3

The POD Method for Evolution Problems

In this chapter, we wish to apply the POD Method to ensembles obtained from the solutions of Evolution Problems at certain time instances (which we shall call "snapshot sets").
In Chapter 4, we shall then use the resulting POD Basis in order to obtain the low-order models for the respective Evolution Problem.

Procedure We show that the theory of Chapter 2 is applicable to ensembles, "generated" by solutions to Evolution Problems.
Furthermore, we investigate the case of "discrete ensembles", i.e., subsets of $\mathbb{R}^m$, which may be given by (finitely many) time-space measurements taken from a numerical simulations of a parabolic IVP for instance.
Then, we focus on the actual calculation of a POD Basis in the case of FE discretizations of Evolution Problems on a "matrix level". Moreover, we review the ingredients of a POD Problem in this context and comment on the particular choices to make. Finally, we carry out an "asymptotic analysis" in the *snapshots*.

Literature The matter is also investigated in detail in the basic lecture notes Volkwein 2006 as well as the diploma thesis Kahlbacher 2006. Especially the results of the FE-case are presented in Volkwein 1999.

3.1 Application of the POD Theory to Evolution Problems

Let us apply the general theory on POD to the case of Evolution Problems. First, we show that the theory is applicable. Then, we derive the statement for the discrete case (which of course is the actual case of interest for numerical applications).

3.1.1 Application to Evolution Problems

In this short subsection we mainly explain that the POD Method is applicable to ensembles obtained from Evolution Problems and introduce some nomenclature for the ingredients of an ensemble in this context.

Recalling the Problem Recall that the Evolution Problem of concern in this thesis (Problem 1.3.2) is of the form

$$\frac{d}{dt}\left(y(t),\varphi\right)_H + a(y(t),\varphi) = \left(F(t),\varphi\right)_H, \quad t \in (0,T) \quad \text{and} \quad y(0) = y_0 \quad \text{in } H, \tag{3.1}$$

which for every $F \in L^2(0,T;H)$ and $y_0 \in V$ (according to Theorem 1.3.3) admits a unique weak solution

$$y \in C([0,T];V) \cap L^2(0,T;D(A)) \cap H^1(0,T;H). \tag{3.2}$$

Concretization of Abstract Ensemble As an ensemble set $\mathcal{V}_P$, we wish to choose solutions of the Evolution Problem at certain time instances and set $\mathcal{V} := \operatorname{span}(\mathcal{V}_P)$. Consequently, the *trajectory* y of the solution presents a *parametrization* of the ensemble and the ensemble parameter set is given by $\Gamma = [0,T]$. Therefore, we shall focus on the *parametrized ensemble* $([0,T], y, \mathcal{V})$.
According to (3.2), we may choose $X = V$ or $X = H$ since X ought to contain the ensemble $\mathcal{V} := \operatorname{span}(\mathcal{V}_P)$. Obviously, the property $y \in C([0,T];X)$ required in Definition 2.1.1 is fulfilled as well. The choice has some influence on the analysis of the resulting Reduced-Order Model (refer to Subsection 4.2.2).

Ensemble Parameter Set for Evolution Problems Let us introduce the specific language for ensembles obtained from Evolution Problems and point out the major difficulty of setting up an actual ensemble. (An ensemble might be "improved" by including the difference quotients of the members or by subtracting the mean of its members; refer to the discussion in Subsection 3.2.1.)

Definition 3.1.1 (Snapshot Grid, Set and Space)
Let $X \in \{V, H\}$ be a separable Hilbert space. Let the members of the ensemble $\mathcal{V}_P \subset X$ consist of the solution $y \in W(0,T)$ of an *Evolution Problem* at certain time instances $0 = t_0 < t_1 < \ldots < t_n = T$. We call $y_j := y(t_j)$, $t = 1, \ldots, n$, *snapshots* and $\mathcal{V}_P := \{y_j\}_{j=1}^n$ a *snapshot set*. We may then define the *snapshot space* to be $\mathcal{V} := \operatorname{span}(\mathcal{V}_P)$. As an (ensemble) parameterization, we choose

$$y : \Gamma \to \mathcal{V}_P, \quad y(t_j) = y_j \quad \text{for } j = 0, \ldots, n,$$

where we call $\Gamma := \{t_j\}_{j=0}^n$ a *snapshot grid* with sizes

$$\delta t := \min\{\delta t_j : 1 \le j \le n\} \quad \text{and} \quad \Delta t := \max\{\delta t_j : 1 \le j \le n\},$$

where $\delta t_j := t_j - t_{j-1} \quad \text{for } j = 1, \ldots, n.$

Remark 3.1.2 (Choice of Snapshot Set)
The choice of the snapshot set is a crucial but also one of the most difficult questions when applying the POD Method.
Theoretically, a POD Basis converges to the "ideal POD Basis" for $\Delta t \to 0$ for just any choice of snapshot grid – as long as the weights α_j in the average operator are chosen adequately (refer to Remark 2.3.6). Yet for practical applications, to the author's knowledge, no reliable techniques have been worked out to choose a proper snapshot grid in a general context (also refer to Subsection 4.3.4).

3.1.2 Application of POD to Discretized Problems

Technically, we consider the same situation as in the previous subsection, but we shall now assume that the ensemble $\mathcal{V}^q \subset \mathbb{R}^m$ is *discrete* and is taken from a discretization of an Evolution Problem at certain time instances (refer to Subsection 1.4 for the case of parabolic IVP).
We shall refine the statements of the POD Problem respectively and shall deduce the solution of the respective POD Problem from the general context of the previous chapter. A "direct" solution for this "discrete case" may be found in detail in Kunisch and Volkwein 1999 as well as in the basic lecture notes Volkwein 2006.

Rephrasing the Objective For a given dimension $\ell \leq n$, our *goal* is to determine a *POD Basis* $\mathcal{B}^\ell$ of rank ℓ that describes best a "snapshot set":

$$\mathcal{V}_P := \{y_j = y(t_j) \mid t_j \in \Gamma_n\} \subset \mathbb{R}^q \quad \text{with} \quad \Gamma_n := \{t_j \in [0,T] \mid j = 1, \dots, n\}.$$

The Snapshot Set – "Coefficient Space" Suppose we "discretize" the Evolution Problem by "approximating" the space X by a q-dimensional space X_h, $q < \infty$. (For parabolic IVPs, we shall construct this by an FE Method.) Choosing a basis for X_h, we obtain a problem in the collection of the respective coefficients (refer to Subsection 1.4.1). Therefore, we choose $X = \mathbb{R}^q$ to be the space of all possible coefficients and our "snapshot space" to be the span of n "snapshots" $y_j \in \mathbb{R}^q$; i.e., we set $\mathcal{V}^q := \operatorname{span}(\mathcal{V}_P) \subset \mathbb{R}^q$ (refer to Definition 3.1.1).

Refinement of Problem In general, the inner product in a "coefficient space" $\mathbb{R}^q$ has to be *weighted* by a *symmetric* matrix $W = [w_1, w_2, \dots, w_q] \in \mathbb{R}^{q \times q}$. (For the case of FE coefficients Proposition 1.4.8 has taught us that we may choose W to be the "mass" or the "stiffness" matrix.) This *weighted* inner product then reads

$$(u, v)_W := u^T W v \quad \text{for all } v, w \in \mathbb{R}^q. \tag{3.3}$$

According to Subsection 2.3.1, in particular (2.23), we calculate the mean by a *weighted* average operator (such that the convergence results derived hold true):

$$\omega_n \in L^2(\Gamma_n), \quad \omega_n(t_j) := \frac{1}{n}\alpha_j, \quad j = 0, \dots, n, \quad \langle F(y(t))\rangle_{t\in\Gamma_n} := \frac{1}{n}\sum_{j=1}^{n} \alpha_j F(y_j),$$

where α_j, $j = 1, \dots, n$, are positive weights. For future use, we define a (*symmetric*) matrix

$$D = \operatorname{diag}(\alpha_1 \dots, \alpha_n).$$

The constant factor $1/n$ will be ignored in the following POD Problem which simply is Problem 2.3.1 for $X := \mathbb{R}^q$ with weighted inner product:

Problem 3.1.3 (POD Problem for Evolution Problems)
Find an *orthonormal* basis $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ that fulfills

$$(\Gamma_n, y, \mathcal{V}^q, P_W^\ell, \langle\cdot\rangle_{t\in\Gamma_n}), \qquad \min_{\mathcal{B}^\ell} J_w := \sum_{j=1}^{n} \alpha_j \left\| y_j - \sum_{k=1}^{\ell} (y_j, \psi_k)_W \psi_k \right\|_W^2 .$$

Link to Abstract Case Essentially, all we need to do is to construct an *analogue* $\mathcal{Y}_n$ of the operator $\mathcal{Y}$, introduced in Proposition 2.2.7. We may then construct the POD operators $R_n := \mathcal{Y}_n\mathcal{Y}_n^*$ and $K_n = \mathcal{Y}_n^*\mathcal{Y}_n$ for this case. It will turn out that $\mathcal{Y}_n$ essentially is given by:

Definition 3.1.4 (Ensemble Matrix)
The *Ensemble Matrix* $Y \in \mathbb{R}^{n\times m}$ is the matrix with rank m, whose columns are the n elements of $\mathcal{V}$.

Solution of Evolution-POD Problem – Need for Transformation By means of $\mathcal{Y}_n$ and results of Section 2.2, we may derive three ways of solving the discrete Evolution-POD Problem 3.1.3. These three possibilities are depicted in Figure 3.1.
Unfortunately, it shall turn out that $\mathcal{Y}_n$ and $\mathcal{Y}_n^*$ are not adjoint in this discrete setting and hence we need to transform the problem in order to make use of the theory of Section 2.2. This transformation is not desired in practical applications, yet at this stage, we wish to enlighten the analogy to the

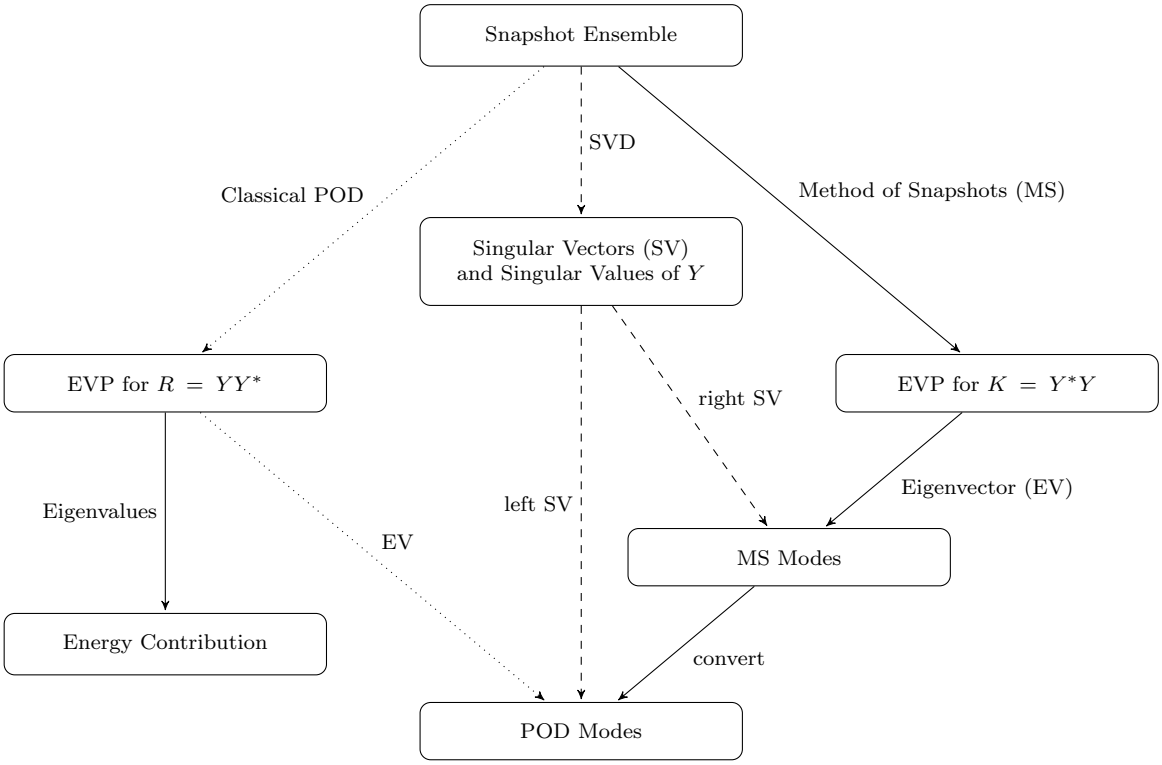

Figure 3.1: Different ways of solving the discrete Evolution-POD Problem 3.1.3

abstract case. In Subsection 3.2.3, we shall then state a simplified version, suitable for practical applications.

Theorem 3.1.5 (Solution of Evolution-POD Problem)
Define

$$\bar{Y} := W^{1/2} Y D^{1/2} \in \mathbb{R}^{m \times m}$$

and proceed in one of the following manners

- Calculate an SVD of $\bar{Y} = U \Sigma V^T$ and let $\{\bar{\psi}_i\}_{i=1}^{\ell}$ consist of the first ℓ columns of U.
- (Classical POD) Find *orthonormal* eigenvectors to the ℓ largest eigenvalues of:

$$R_n' \bar{\psi}_i = \bar{Y} \bar{Y}^T \bar{\psi}_i = \lambda_i \bar{\psi}_i \quad \text{in} \quad \mathbb{R}^q.$$

- (Method of Snapshots) Find *orthonormal* eigenvectors to the ℓ largest eigenvalues of:

$$K_n' \bar{u}_i = \bar{Y}^T \bar{Y} \bar{u}_i = D^{1/2} Y^T W Y D^{1/2} \bar{u}_i = \lambda_i \bar{u}_i \quad \text{in} \quad \mathbb{R}^n$$

$$\text{and set} \quad \bar{\psi}_i := \frac{1}{\sqrt{\lambda_i}} \bar{Y} \bar{u}_i.$$

Then, the solution to Problem 3.1.3, i.e., the *POD Basis of rank l* for this case, is given by $\mathcal{B}^{\ell} = \{\psi_i\}_{i=1}^{\ell}$ which consists of

$$\psi_i = W^{-1/2} \bar{\psi}_i, \quad i = 1, \ldots, \ell.$$

Proof.
In context of discretized Evolution Problems, we derive the form $\mathcal{Y}_n$ of the operator $\mathcal{Y}$ (defined in Proposition 2.2.7) and interpret it on a *matrix level*. Analogously, we proceed with $\mathcal{Y}^*$. We then setup the respective operators R_n and K_n by means of Proposition 2.2.7 and Proposition 2.2.10, respectively.
Finally, we transform the EVP for R_n on a *matrix level* to a problem for a matrix R_n' in order to derive a matrix $\bar{Y}$ such that R_n' might be written as $R_n' = \bar{Y}\bar{Y}^T$.
Throughout, let $v^{(i)}$ denote the i-th component of $v \in \mathbb{R}^n$.

• Derivation of $\mathcal{Y}_n$ Note that from the definition of $\mathcal{Y}$ in Proposition 2.2.7 (for "trivial" weights) we infer by (2.2)

$$\mathcal{Y} : L^2(\Gamma_n) \to \mathbb{R}^q, \quad \mathcal{Y}w = (w, y)_{L^2(\Gamma_n)} = \sum_{j=1}^{n} w(t_j)y(t_j) \quad \text{for } w \in L^2(\Gamma_n).$$

We represent $w \in L^2(\Gamma_n)$ by $v \in \mathbb{R}^n$ with $v^{(j)} = w(t_j)$. Then, introducing the weights α_j, we obtain the analogous operator

$$\mathcal{Y}_n : \mathbb{R}^n \to \mathbb{R}^q, \quad (\mathcal{Y}_n v)^{(k)} = \sum_{j=1}^{n} \alpha_j v^{(j)} y_j^{(k)}, \quad k = 1, \dots, q, \quad v \in \mathbb{R}^n.$$

Thus, for the standard basis $\{e_k\}_{k=1}^n$ of $\mathbb{R}^n$ we obtain

$$\mathcal{Y}_n e_k = \sum_{j=1}^{n} \alpha_j \underbrace{e_k^{(j)}}_{=\delta_{jk}} y_j = y_k \alpha_k \in \mathbb{R}^q,$$

which for the matrix representation in the respective basis (and the definition of D) implies that $\mathcal{Y}_n$ is given in terms of the Ensemble Matrix Y of Definition 3.1.4:

$$\mathrm{Matrix}_{\{e_k\}_{k=1}^n}^{\{e_k\}_{k=1}^q}(\mathcal{Y}_n) = [y_1, y_2, \dots, y_n]D = YD.$$

• Derivation of $\mathcal{Y}_n^*$ By the definition of $\mathcal{Y}^*$ in Proposition 2.2.7, we have

$$\mathcal{Y}^* : \mathbb{R}^q \to L^2(\Gamma_n), \quad (\mathcal{Y}^* z)(t_j) = (y(t_j), z)_W = (y_j, z)_W, \quad j = 1, \dots, n.$$

Representing $\mathcal{Y}_n^* z \in L^2(\Gamma_n)$ by $\mathcal{Y}_n^* z \in \mathbb{R}^n$ such that there holds $(\mathcal{Y}_n^* z)(t_j) = (\mathcal{Y}_n^* z)^{(j)}$, we obtain the analogue

$$\mathcal{Y}_n^* : \mathbb{R}^q \to \mathbb{R}^n, \quad (\mathcal{Y}_n^* z)^{(j)} = (y_j, z)_W = y_j^T W z = \sum_{i=1}^{q} y_j^{(i)} (Wz)^{(i)}, \quad j = 1, \dots, n.$$

Applied to the standard basis $\{e_k\}_{k=1}^n$ of $\mathbb{R}^q$, this reads

$$(\mathcal{Y}_n^* e_k)^{(j)} = \sum_{i=1}^{q} y_j^{(i)} (We_k)^{(i)} = \sum_{i=1}^{q} y_j^{(i)} (m_k)^{(i)} = (Y^T W)_{kj}, \quad j = 1, \dots, n.$$

In vector form, this yields

$$\mathcal{Y}_n^* e_k = (Y^T W)_{k\cdot} \quad \text{for} \quad j = 1, \dots, n,$$

which implies for the matrix representation (w.r.t. the standard basis):

$$\mathrm{Matrix}_{\{e_k\}_{k=1}^n}^{\{e_k\}_{k=1}^q}(\mathcal{Y}_n^*) = Y^T W.$$

• Operators R_n and K_n Let us now setup the corresponding operators R_n and K_n by means of Proposition 2.2.7 and Proposition 2.2.10, respectively:

$$R_n := \mathcal{Y}_n \mathcal{Y}_n^* = YDY^T W, \quad K_n := \mathcal{Y}_n^* \mathcal{Y}_n = Y^T WYD. \tag{3.4}$$

Proposition 2.2.10 also yields a "conversion formula" of eigenvectors u of K_n to eigenvectors ψ of R_n (to the eigenvalue λ):

$$\psi = \frac{1}{\sqrt{\lambda}} \mathcal{Y}_n u = \frac{1}{\sqrt{\lambda}} YDu. \tag{3.5}$$

• Transformation of EVP Obviously, $\mathcal{Y}_n$ are $\mathcal{Y}_n^*$ not "$\mathbb{R}^n$-$\mathbb{R}^q$-adjoint" (i.e. "transpose") of each other. Thus, we infer $R_n \neq \mathcal{Y}_n \mathcal{Y}_n^T$.
Let us transform the EVP for R_n to a problem for a matrix R_n', that we may write as $R_n' = \bar{Y}\bar{Y}^T$ with a suitable matrix $\bar{Y}$. In particular, we introduce the "*transformation*"

$$\bar{\psi}_k := W^{1/2} \psi_k$$

and insert $\psi_k = W^{-1/2}\bar{\psi}_k$ into the EVP for R_n

$$R_n \psi_k = YDY^T W \psi_k = \lambda_k \psi_k.$$

Additionally, we multiply the equation by $W^{1/2}$ (from the left) to obtain

$$(W^{1/2} YDY^T W) W^{-1/2} \bar{\psi}_k = W^{1/2} \lambda_k W^{-1/2} \bar{\psi}_k,$$

which yields an EVP for $R_n' := W^{1/2} \mathcal{Y}_n \mathcal{Y}_n^* W^{1/2}$:

$$R_n' \bar{\psi}_k = W^{1/2} YDY^T W^{1/2} \bar{\psi}_k = \lambda_k \bar{\psi}_k.$$

• Decomposition of R_n' Since $D^T = D$ and $W^T = W$, this holds for their roots and we may define

$$\bar{Y} := W^{1/2} YD^{1/2} \quad \text{such that} \quad \bar{Y}^T = D^{1/2} Y^T W^{1/2},$$

which together decompose R_n' as well as K_n' (which might be shown similarly):

$$R_n' = \bar{Y}\bar{Y}^T \quad \text{and} \quad K_n' = \bar{Y}^T \bar{Y}.$$

• Completing the Proof It remains to re-transform $\psi_k = W^{-1/2}\bar{\psi}_k$ for $k = 1, \ldots, \ell$. The result on the SVD follows from Remark 1.1.4 on the formulation of an SVD as an EVP. □

3.2 Finding a POD Basis for Reducing FE Models

In this section, we apply the POD Method in a practical context, i.e. we make use of snapshot sets obtained from FE discretization of Evolution Problems. The main objective of course being, to use the POD Basis in Reduced-order Modeling (refer to Chapter 4).

Procedure We discuss all ingredients of a POD Problem (refer to Subsection 2.1.1) and give hints on improving the snapshot set. Then, we investigate the appearance of the POD Method when the snapshots are obtained from an FE simulation and summarize the procedure. We close by carrying out an "asymptotic analysis" in the snapshots as well as looking at the numerical properties of the POD operator.

3.2.1 Improving the Snapshot Set for Evolution Problems

Bearing in mind Remark 3.1.2 on the difficult choice of snapshots, we discuss ways to improve a given snapshot set.

Mean Subtraction from Snapshots A first way to improve a given snapshot set is to subtract the mean of the snapshot from the snapshot set. Then, the snapshot set only consists of the "fluctuations" of the snapshots. This subtraction may be important if the magnitude of the fluctuations is small in comparison to the magnitude of the mean. If the mean is not subtracted in this case, the fluctuations would not be captured appropriately since they have only little "relevance" in comparison to the mean component.
Furthermore, the subtraction may reduce the order of the POD Basis by one (refer to the numerical study in Subsection 6.1.4). We comment on this matter from a theoretical point of view in Chapter A. Yet let us at least "enlighten" the idea: We way obtain a POD Basis by means of an SVD (refer to Theorem 3.1.5). Recall the *geometric interpretation* of the SVD (Subsection 1.1.1). By subtracting the mean of each snapshot the "cloud" of all snapshots is shifted to zero. Furthermore, the POD Method then produces a basis for a linear subspace (rather than an "affine" one). So for the same quality of approximation this generally will reduce the number of basis elements needed by one. For an illustration of this procedure, refer to Chatterjee 2000, Figure 3.

Including "Difference Quotients" into the Snapshot Set We may also add the *finite difference quotients* of the snapshots $\bar{\partial}_t y_j = (y_j - y_{j-1})/\delta t_j$ to the snapshot set $\mathcal{V}_P = \{y_j\}_{j=0}^n$, introduced in Definition 3.1.1 and obtain:

$$\mathcal{V}_P' := \{y_j\}_{j=0}^n \cup \{\bar{\partial}_t y_j\}_{j=1}^n. \tag{3.6}$$

$\mathcal{V}_P'$ surely is linearly dependent which does not constitute a difficulty to the method: The resulting POD Basis will be "properly orthogonal", i.e., linearly independent for all snapshot sets chosen. But let us point out that the resulting POD Basis *depends* on whether it is obtained using $\mathcal{V}_P$ or $\mathcal{V}_P'$.
In Subsection 4.2.2, we shall see that by means of $\mathcal{V}_P'$, we may improve the error estimates for reduced-order models: The time derivate in problem (1.5) has to be approximated. It turns out that we will obtain error estimates which do not depend "badly" on the snapshot grid size Δt if we use an "extended" snapshot set $\mathcal{V}_P'$ (also refer to Kunisch and Volkwein 2001, Remark 1).

3.2.2 POD Strategies – Choices to Make in a Practical Context

In order to clarify the various choices to be made when applying the POD Method, we shall list them again and comment on their respective features.

Parameterization of the Ensemble – Classical Approach vs Method of Snapshots We have obtained our snapshot set from an Evolution Problem which was discretized in *"vertical"* fashion. In Chapter 4, we wish to establish a (model) reduction of the system of ODEs obtained from the (spatial) Galerkin approach (refer to Proposition 1.4.2). Hence, we look for "key *spatial* structures" on a "time average". In terms of the nomenclature introduced in Subsection 2.1.1, we have thus chosen the "ensemble parameter" to be the time.
In terms of actually *calculating* a POD Basis, it might however be handy to *act* as if our ensemble was taken in space and obtain key temporal structures, which might then be *converted* to the desired spatial structures.
In particular, if the snapshots are taken from a physical experiment, we naturally have got lots of "measurements" in time at comparably few locations in space. Hence, it is advisable to chose the respective *time* instances as an ensemble parameter, which leads to a smaller problem to solve in order to establish a POD Basis (due to $q \ll n$ and Theorem 3.1.5). This approach is called *"Classical POD"*.
On the other hand, if the measurements are taken from a numerical simulation, we usually are given lots of measurements in space at comparably few time instances. In this case, choosing the *space* points as an ensemble parameter, leads to a smaller problem to solve (Theorem 3.1.5; $n \ll q$). Having obtained snapshots in time, we therefore have to *act* as if we were given snapshots in space (*"spaceshots"* so to say), calculate key *temporal* structures and transform them into the key *spatial* structures of desire. This procedure is called *Method of Snapshots* and was first suggested in Sirovich 1987. (This method is established in Theorem 3.1.5, yet in Chapter A, we give a justification of the method from another point of view: POD as a "Bi-orthogonal Decomposition")

The Optimality Norm – Choice of Projection We also need to define the norm in which the approximation of the ensemble should be optimal. (According to Remark 2.1.7, the Optimality Norm and the POD Projection are linked by the inner product. Hence, the following considerations also influence the choice of the POD Projection.)
Due to Definition 3.1.1, we may choose $X = V$ or $X = H$, i.e., we may choose $\|\cdot\|_V$ or $\|\cdot\|_H$ in the context of Evolution Problems. For the case of "parabolic initial value problems of second order", we have chosen $V := H_0^1(\Omega)$ and $H := L^2(\Omega)$ (refer to Problem 1.3.6). If we additionally apply an FE discretization say, we may construct analogues of these norms by means of the "mass" or the "stiffness" matrix (refer to Proposition 1.4.8).
From a physical point of view, $L^2(\Omega)$ consists of all functions of "finite energy". Thus, if we would like our POD modes to represent the components of highest "energy" we would have to chose $\|\cdot\|_{L^2(\Omega)}$ as an Optimality Norm. (The notion of energy is only applicable in certain contexts though; refer to the discussion in Subsection 4.3.2).
On the other hand, it will turn out in Chapter 4, that the error bounds of reduced-order models may be better controlled when using the H^1-norm for example (refer also to Kunisch and Volkwein 2002, Theorem 4.7). Note however that this norm is more expensive to compute.

The Averaging Operator In Section 2.3, we proposed to use a *weighted* average operator (in particular, a trapezoidal approximation of an integral for example). This ensured, for the number of snapshots n approaching infinity, the POD Basis $\mathcal{B}_n^\ell$ to converge to an *"ideal" POD Basis* $\mathcal{B}_\infty^\ell$.
This convergence property shall enable us to present error estimates for Reduced-order Modeling which are independent of the actual choice of the *snapshot grid* (Refer to Subsection 4.2.2).
(In context of Evolution Problems, $\mathcal{B}_\infty^\ell$ is obtained for the ensemble set being the interval $[0, T]$, i.e., by taking into account *all* snapshots possible. A POD Basis $\mathcal{B}_n^\ell$, obtained from a snapshot of finitely many snapshots, is only optimal for the specific choice of snapshot set; refer to Remark 4.3.1.)

Choice of ℓ – Sequence of subspaces From the point of view of "pure mathematics", by varying ℓ, we have constructed a *sequence* of linear finite-dimensional subspaces $X^\ell := \operatorname{span}(\mathcal{B}^\ell)$ such that each one is optimal at the respective dimension ℓ (in terms of representing the ensemble $\mathcal{V}$).

In practical applications, we usually are only interested in the optimal representation for one *particular* value of ℓ, however. A suitable "rule of thumb" is to choose ℓ in such a way that a certain percentage of the "energy" in the snapshots is captured (refer to the warning in Subsection 4.3.2).
Note that this percentage only teaches us something about the representation of the *snapshots* rather than of the *system itself* (refer to Remark 4.3.1). The percentage of energy represented is usually chosen to be 95% or 99% and may be calculated by

$$\mathcal{E}(\ell) = \frac{\sum_{i=1}^{\ell} \lambda_i}{\sum_{i=1}^{q} \lambda_i}\,.$$

In general, we expect the first eigenvalues to *decay exponentially*, therefore we expect ℓ to be reasonably *low*. (Experimental studies on the "order of decay" have been carried out for various cases – for example in Kahlbacher 2006, Subsection 3.5.2.)

3.2.3 The POD Method and FE-Discretizations

In this subsection, we shall explicitly investigate how to calculate a POD Basis for snapshot sets obtained from FE discretizations (of parabolic IVP say).
Since an FE space is isomorphic to $\mathbb{R}^q$ (Proposition 1.4.8), mathematically speaking, this subsection covers a special case of Subsection 3.1.2 – for particular choices of $\mathcal{V}^q$, $(\cdot,\cdot)_W$ as well as the weights $\{\alpha_j\}_{j=1}^n$. Anyhow, we wish to give these particular choices and state the problem on an "FE matrix level".

Results from FE Theory For the Hilbert space X, we choose an FE space $X_{\mathcal{T}}$ (see Proposition 1.4.6). By the choice of FE ansatz functions $\{\varphi_i\}_{i=1}^q$ for $X_{\mathcal{T}}$ and Proposition 1.4.8, we may find that for $q \in \mathbb{N}$ being the number of degrees of freedom

$$X^h \cong \mathbb{R}^q. \tag{3.7}$$

Therefore, according to (1.20), we may represent each $y^h \in X_{\mathcal{T}}$ by a corresponding FE vector $c \in \mathbb{R}^q$ by writing (with $c^{(i)}$ denoting the i-th component of c):

$$y^h = \sum_{i=1}^{q} c^{(i)} \varphi_i.$$

Setup of the FE-POD Problem Let the ensemble $\mathcal{V}^q$ consist of all these FE vectors of snapshots of a FE solution to a dynamical system (such as 1.5):

$$\mathcal{V}^q := \{c_j\}_{j=1}^q \subset \mathbb{R}^q.$$

According to the actual choice of X_h and Proposition 1.4.8, we choose W in the definition of the inner product (3.3) to be a matrix such as the mass- or the stiffness matrix. (For finite difference schemes we would choose $W = \mathrm{diag}(h/2, h, ..., h, h/2)$ for example.)
In terms of average operator, we choose the weights α_j such that the problem can be considered to be a trapezoidal approximation of the time continuous case and hence, convergence is ensured:

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

where δt_j, $j = 1, \dots, n$, are the grid sizes defined in Definition 3.1.1.
Then, the corresponding POD Problem is exactly given by Problem 3.1.3 with the particular choices for the weights $\{\alpha_j\}_{j=1}^q$ and the matrix W.

Solution of the FE-POD Problem Obviously, as the problem is just a special case of Problem 3.1.3, the solution is immediately given by Theorem 3.1.5 where in particular, we shall use the *Method of Snapshots* as we will naturally obtain way more FE coefficients than snapshots (see the previous subsection). Note that K^h is given in a simplified form of K'_n. (Let us explicitly indicate the dependence of the snapshots on the mesh size h as we wish to investigate this dependence later on.)

Corollary 3.2.1 (Solution of FEM POD Problem)
Gather all snapshots (FE vectors) as columns in an *Ensemble Matrix* $Y \in \mathbb{R}^{q\times n}, Y_{ij} = c_j^{(i)}$ with

$$y_j^h = \sum_{i=1}^{q} Y_{ij}^h \varphi_i, \quad j = 1, \dots, n. \tag{3.8}$$

Solve the eigenvalue problem

$$K^h u_k := (Y^h)^T M^h Y^h D u_k = \lambda_k u_k \quad \text{for } k = 1, \dots, \ell, \tag{3.9}$$

where M^h is the mass matrix introduced in Definition 1.4.7 and D denotes the matrix of the weights of the average operator. Let the eigenvalues λ_k be in decreasing order and let u_i denote the corresponding eigenvector.
The POD Basis $\mathcal{B}^\ell$ of rank ℓ is then given by:

$$\psi_i := \frac{1}{\sqrt{\lambda_i}} Y D u_i \qquad \text{for } i = 1, \dots, \ell.$$

Proof.
This is an immediate consequence of Theorem 3.1.5 for the alternative "Method of Snapshots" and the choices made above. In particular, due to (3.4), there holds $K^h = K_n$. □

Summarized Procedure of Calculating a POD Basis Let us summarize our findings in a simple "algorithm" for obtaining a POD Basis from snapshots generated by an FE-discretized Evolution Problem by means of the Method of Snapshots:

1. Calculate n snapshots y_j, $1 \dots, n$, by the FE method.
2. "Improve" the snapshot set: subtract mean, add difference quotients if desired.
3. Gather the snapshots (FE vectors) in an Ensemble Matrix $Y = [y_1, \dots, y_n]$ "column-wise".
4. Build the matrix correlation operator $K = Y^T M Y D$ where $D = \operatorname{diag}(\alpha_1, \dots, \alpha_n)$ ("trapezoidal weights").
5. (Iteratively,) obtain the ℓ largest eigenvalues $\{\lambda_i\}_{i=1}^{\ell}$ and the corresponding eigenvectors $\{u_i\}_1^{\ell}$ of K.
6. For $i = 1, \dots, \ell$, transform u_i into a POD mode $\psi_i := \frac{1}{\sqrt{\lambda_i}} Y D u_i$.

3.2.4 Asymptotic Analysis of Snapshots and Numerical Properties

Since the role of K^h is central to calculating a POD Basis, we shortly comment on its numerical properties. In analogy to Section 2.3, we wish to carry out an "asymptotic analysis" in the *snapshots* (instead of the *snapshot grid*). We carry out a *numerical* study on that matter in Subsection 6.2.5.

Remark on Temporal Asymptotic Analysis Note that we shall not touch upon the issue of a *temporal* "asymptotic analysis" in the snapshots. This would be necessary since in the asymptotic analysis in Section 2.3, we only considered the choice of the snapshot grid and assumed the corresponding snapshot set to be known exactly. Yet, in practice, it does matter whether we obtain a snapshot set (on a given snapshot grid) from a "coarse" time grid of a solution or a fine one (also refer to the numerical example in Subsection 6.2.3).

The Setting As a reference, we consider a Hilbert space X (just as in Chapter 2). We consider the "exact" operator K obtained from the "exact" snapshot ensemble

$$\mathcal{V} := \{y_j\}_{j=1}^n \subset X.$$

In (3.9), we have introduced K^h, constructed from an "approximated" snapshot ensemble lying in some FE space $X_{\mathcal{T}}$, characterized by the "mesh size" h:

$$\mathcal{V}^h := \{y_j^h\}_{j=1}^n \subset \mathbb{R}^q \cong X_{\mathcal{T}}.$$

(From now on, we may however think of *any* finite dimensional approximation X^h to X parametrized by h, not necessarily arising from an FE discretization.)

Spatial Asymptotic Analysis of Snapshots In Section 2.3, we have investigated the behaviour of the POD Basis for the snapshot grid to converge to the continuous solution interval, i.e., we have carried out an "asymptotic analysis" in the ensemble parameter, i.e., in time. In this context, we have assumed that the snapshots are known *exactly*. Obviously this is not the case in a practical context since the snapshots would be taken from a (discrete) numerical simulation say. In particular, we expect perturbations in time and space, where we concentrate on the latter case.
Therefore, we now wish to carry out a "spatial" asymptotic analysis, i.e., we wish to investigate how the POD operator K behaves when a "discrete" snapshot set $\mathcal{V}^h$ converges to the reference set $\mathcal{V}$ – for the same choice of snapshot grid. In particular, we investigate the convergence properties of K^h to K for X^h converging to X by h approaching zero.
Analogously to Subsection 2.3.2, we should then think about the convergence of the resulting POD Basis, i.e., the eigenvectors of K^h. Yet this unfortunately is beyond the scope of this thesis.
Therefore, let us concentrate on the actual finding in terms of convergence of the operators. (Note that property (3.10) for FE approximations is ensured by convergence results for FE spaces such as (1.19).)

Proposition 3.2.2 (Convergence Properties of $K(h)$)
Let X be a separable Hilbert space. For a family $\{h\}_{h>0}$ with accumulation point zero, we define X^h to be an FE space. Furthermore, let Π^h be the bounded, $(\cdot,\cdot)_X$-orthogonal projection from X onto X^h. The members of the snapshots sets $\mathcal{V} = \{y_j\}_{j=1}^n$ and $\mathcal{V}^h = \{y_j^h\}_{j=1}^n$ may then be "connected" by writing

$$y_j^h = \Pi^h y_j \in X^h \qquad \text{for } j = 1, \dots, n.$$

For each h, we introduce the operator K^h according to (3.9) and define

$$K(h) := \begin{cases} K^h & \text{for} \quad h > 0, \\ K & \text{for} \quad h = 0. \end{cases}$$

Then, there holds:

- If the family of restrictions $\{\Pi^h\}_{h>0}$ is point-wise convergent in X, i.e.,

$$\lim_{h\to 0} \Pi^h u = u \quad \text{for any} \quad u \in X, \tag{3.10}$$

then, $K(h)$ is right-continuous at $h = 0$, i.e.,

$$\lim_{h \searrow 0} K(h) = K.$$

- If in addition there exists $\epsilon > 0$ such that

$$\max_{1 \le j \le n} \left\| \Pi^h y_j - y_j \right\|_X = O(h^\epsilon) \quad \text{for} \quad h \to 0,$$

then,

$$\|K - K(h)\|_2 = O(h^\epsilon) \quad \text{for} \quad h \to 0,$$

where $\|\cdot\|_2$ denotes the spectral norm for matrices. I.e., the order of the approximation of the snapshots coincides with the order of approximation of the respective correlation matrices K^h.

Proof.
Refer to Volkwein 1999, Proposition 6. □

Properties for the Choice of Algorithm For quite a number of numerical algorithms to solve eigenvalue problems, the following properties of the respective operator are of importance. (The remark is an excerpt of Volkwein 1999, Remark 6.)

Remark 3.2.3 (Properties of K and K^h)
The matrix K is symmetric and positive definite. The matrix K_h is symmetric and positive *semi*-definite. If (3.10) holds true, K_h becomes positive definite for sufficiently small h (due to the convergence result of Proposition 3.2.2).

"Sensitiveness" of the POD solution We wish to comment on the "sensitiveness" of the POD basis (basically, the eigenvectors of K and K^h) on the given data by quoting a result on the respective condition numbers of K. (A relation to the eigenvectors of K^h is given in Volkwein 1999, Theorem 8 which was proved in Demmel 1997.)
Roughly speaking, eigenvectors of K are "sensitive" (or their condition number is large, i.e., their condition is "bad") if the gap of the corresponding eigenvalue to the nearest other eigenvalues is small. Therefore, close eigenvalues lead to large condition numbers of the corresponding eigenvectors.

Proposition 3.2.4 (Condition of Eigenvalues and Eigenvectors of K)
The condition number of an eigenvalue λ of the matrix K is 1. The condition number of the corresponding eigenvector v is given by

$$\mathrm{cond}(v) = \frac{1}{\min_{\mu \in \sigma(K) - \{\lambda\}} |\mu - \lambda|}, \tag{3.11}$$

where $\sigma(K)$ denotes the spectrum of K.

Proof.
The result on the condition number of the eigenvalue is given in Volkwein 1999, Theorem 7. A proof of (3.11) may be found in Chatelin 1983. □

Chapter 4

Reduced Order Modeling for Evolution Problems

In this chapter, we shall use the POD Method to find a low order approximation for the Evolution Problem 1.3.2. We call this procedure "*Reduced Order Modeling*" (ROM). In Section 5.4, we shall then use these "low-dimensional" models to develop respective Suboptimal Control strategies.

Procedure We shortly explain the idea of Model Reduction and introduce POD reduced-order models for Evolution Systems as a special sort of Galerkin approach. We establish their formulations for the "time-continuous" as well as the time discrete case. Then, we consider POD reduced-order models for FE discretized systems and show how the "FE system" and the "low-order POD system" are linked.
We then carry out a thorough error analysis of POD reduced-order models and conclude with a discussion of the POD Method as a technique for Model Reduction.

Literature The theory presented may in most parts be found in Kunisch and Volkwein 2002 as far as the error analysis is concerned and in Kunisch and Volkwein 2001 in terms of little remarks and extensions. Some elements of the discussion are taken from Chatterjee 2000.

4.1 POD Reduced-Order Models

In this section, we shall apply a low-order basis of snapshots obtained by the POD Method in order to obtain a low-dimensional model for the Evolution Problem 1.3.2.

Procedure We give an intuitive idea of a reduced-order model. POD Reduced-Order Modeling is based on a Galerkin approach. Hence, we derive the model from the general Galerkin approach of Subsection 1.4.1 for the time continuous as well as the time discrete case of Evolution Problems. Then, we focus on setting up a reduced-order model for an FE discretization.

The Model of Concern As mentioned before, throughout this thesis, we consider the Evolution Problem 1.3.2 whose essentials we herewith repeat for the sake of convenience:
Let $V \subset H \subset V^*$ be a Gelfand triple. (In particular, we due to the continuous embedding have $\|\cdot\|_H \leq c_V \|\cdot\|_V$.) We seek a *generalized solution* $y \in W(0,T)$ such that there holds

$$\frac{d}{dt}(y(t),\varphi)_H + a(y(t),\varphi) = (F(t),\varphi)_H \quad \text{for all} \quad \varphi \in V, \quad t \in (0,T], \tag{4.1a}$$

$$y(0) = y_0 \in H. \tag{4.1b}$$

Further Prerequisites Suppose that, based on a certain snapshot set, we have determined a POD Basis $\{\psi_k\}_{k=1}^{\ell}$ of rank ℓ for some fixed $\ell \in \mathbb{N}$. Recall that we set $\mathcal{V}^\ell := \operatorname{span}(\psi_1, \dots, \psi_\ell)$. According to (4.1), we seek a solution $y \in W(0,T)$. Therefore, we may assume $y(t) \in V$ for all $t \in [0,T]$.
Hence, we set $X := V$ in the POD Problem. (We shall consider the case $X := H$ in Subsection 4.2.3 only briefly.) Note that this choice actually determines the "Optimality Norm" in the POD Method: We consider reduced-order models based on a POD Basis which *in the X-norm* optimally represents "snapshots" of the system (refer to Remark 2.1.7 on the Optimality Norm).

4.1.1 Introductory Remarks on Model Reduction

Let us explain the basic idea of Model Reduction and show the connection to (POD) Galerkin systems.

Variants of Model Reduction Techniques The general interest of Model Reduction of course is to *reduce the numerical effort* to compute a solution to a "model", such as an Evolution Problem. Especially "optimization problems with PDE constraints" lead to large problems which we desire to reduce in size.
In this thesis, we shall exclusively consider the *POD Method*, yet of course there is quite a variety of other approaches. A good overview is given in the textbook Antoulas 2005. (In particular, a categorization of model reduction methods is depicted in Figure 1.3 therein.) We learn that POD is an SVD-based projection method (in contrast to Krylov Methods) which is also applicable to *non-linear* systems. (In this thesis however, we shall apply it to *linear* systems only.)

Low-order Approximation Using the approach of Subsection 2.2.3, we may say that a discretization of an Evolution leads to an *approximation*, whose coefficients are to be determined by a "discrete" model. Since the solution space V is separable, we may write for an orthonormal basis $\{\psi_k\}_{k\in\mathbb{N}}$ of a dense subspace of V:

$$y(t) = \sum_{k=1}^{\infty} (y(t), \psi_k)_X \, \psi_k \quad \text{for all } t \in [0,T].$$

Carrying out a Galerkin approach with *orthonormal* POD modes as ansatz functions, we obtain an approximation for $y(t)$:

$$y^\ell(t) = P^\ell y(t) = \sum_{k=1}^{\ell} (y(t), \psi_k)_X \, \psi_k, \tag{4.2}$$

where for all $t \in [0,T]$, the coefficients $y_k^\ell(t) := (y(t), \psi_k)_X$ remain to be determined.
A similar procedure leads to an "FE discretization" of the Evolution Problem (with q degrees of freedom). We expect that in general $\ell \ll q$, i.e., that we have obtained a *low-dimensional* approximation of the Evolution Problem (in comparison to an FE discretization).

Galerkin Formulation Let us deduce from the considerations above that the size of the resulting Galerkin system is reduced: As seen in Subsection 1.4.1, by means of Galerkin ansatz functions $\{\varphi_i\}_{i=1}^{q}$, we obtain a linear system of ODEs, which for matrices D and A and a suitable RHS F reads (refer to (1.14))

$$D\frac{d}{dt}c + Ac = \hat{F}, \qquad y(0) = y_0. \tag{4.3}$$

The solution of this IVP corresponds to the coefficients of the Galerkin solution w.r.t. the Galerkin ansatz functions, i.e., our solution space is $V_h = \mathbb{R}^q$ where q denotes the number of ansatz functions. This implies that the "size" of this system depends on the dimension of the "test space" V_h.
In terms of parabolic IVPs, we choose an FE Space $X_{\mathcal{T}}$ as a test space $V_h := X_{\mathcal{T}}$. On the other hand, choosing $V_h := \mathcal{V}^\ell$ as a test space, we expect to obtain a way smaller system since we expect $\ell \ll q$ where q denotes the number of degrees of freedom in the FE system.

4.1.2 The Galerkin POD Method

We introduce a variant of the Galerkin approach of Subsection 1.4.1 and find the resulting linear system. This subsection presents an introduction of the respective notation and a corresponding refinement of the Galerkin approach rather than actually establishing new results.

Abstract POD Galerkin Method In the Galerkin approach for an Evolution Problem (Problem 1.4.1), we choose V_h to be $\mathcal{V}^\ell$ and denote the corresponding solution by y^ℓ instead of y_h. Since $a(\cdot,\cdot)$ is a symmetric and coercive bilinear form and $\mathcal{V}^\ell$ is of finite dimension, it follows that there exists a unique solution y^ℓ to the resulting "POD Galerkin approach" (refer also to Hinze and Volkwein 2005, Proposition 3.4):

Problem 4.1.1 (Galerkin POD Problem)
Find a function $y^\ell \in C([0,T];\mathcal{V}^\ell)$ such that

$$\frac{d}{dt}\left(y^\ell(t),\psi\right)_H + a(y^\ell(t),\psi) = (f(t),\psi)_{V',V} \quad \text{for } \psi \in \mathcal{V}^\ell,\ t \in (0,T], \tag{4.4a}$$

$$\left(y^\ell(0),\psi\right)_H = (y_0,\psi)_H \quad \text{for } \psi \in \mathcal{V}^\ell. \tag{4.4b}$$

POD Low-Order System Just as in Subsection 1.4.1, we choose a basis for the ansatz space $\mathcal{V}^\ell$ in the Galerkin POD Problem 4.1.1. Due to the definition of $\mathcal{V}^\ell$, we may use a POD Basis. We may then obtain a system of ODEs for the coefficients of the (low-order) solution w.r.t. this basis. For choosing the basis to be a POD Basis, we have to refine the "general" problem matrices and vectors (1.13) and (1.16) (note the usual inversion in the indices). We then obtain an analogue of Proposition 1.4.2 for POD Galerkin systems:

Corollary 4.1.2 (POD Low-Order System)
Let $\{\psi_i\}_{i=1}^\ell$ be a POD Basis. For the low-order solution $y^\ell \in C([0,T];\mathcal{V}^\ell)$, we make the ansatz

$$y^\ell(t) := \sum_{j=1}^{\ell} c_j^\ell(t)\psi_j, \qquad y^\ell(0) = \sum_{j=1}^{\ell} \alpha_j^\ell \psi_j. \tag{4.5}$$

Then, we define

$$\begin{aligned} M^\ell &:= ((\psi_j,\psi_i)_H)_{i,j=1}^q \quad A^\ell := (a(\psi_j,\psi_i))_{i,j=1}^q, \\ F^\ell &:= ((F(t),\psi_j)_H)_{j=1}^q, \quad g^\ell := (\alpha_j^\ell)_{j=1}^q \end{aligned} \tag{4.6}$$

such that we may obtain the coefficients $c^\ell(t) := (c_j^\ell(t))_{j=1}^\ell \in \mathbb{R}^\ell$, $t \in [0,T]$, of the low-order solution w.r.t. the POD Basis from the *"POD low-order system"*:

$$M^\ell \frac{d}{dt} c^\ell(t) + A^\ell c^\ell(t) = F^\ell(t), \qquad y^\ell(0) = g^\ell. \tag{4.7}$$

Proof.
This is a direct consequence of Proposition 1.4.2 for the choice of $V_h := \mathcal{V}^\ell$. □

Remark 4.1.3 (Simplification due to Orthonormality)
Since the POD Basis is X-orthonormal, for $X = H$, M^ℓ becomes the identity matrix and for $X = V$, A^ℓ becomes the identity (recall that in (1.3) we set $(\cdot,\cdot)_V := a(\cdot,\cdot)$). Hence, we may simplify system (4.7) even further.

4.1.3 The Backward Euler Galerkin POD Method

It remains to discretize the Galerkin System in time for which we again use the *implicit Euler* method. (For similar results with the *Crank Nicholson* or the *explicit* Euler method see Kunisch and Volkwein 2001, for example.)

The Time Grid Λ^ℓ In analogy to Λ (Subsection 1.4.2), we introduce a time grid Λ^ℓ for the low order-solution based on a POD Basis of rank ℓ. This basis was obtained from snapshots taken on the grid Γ (which generally is well different from Λ^ℓ). We choose Λ^ℓ to consist of $m \in \mathbb{N}$ time steps:

$$\Lambda^\ell := \{\tau_j\}_{j=0}^m, \quad 0 = \tau_0 < \tau_1 < \ldots < \tau_m = T, \quad \delta\tau_j := \tau_j - \tau_{j-1} \text{ for } j = 1, \ldots, m$$

and additionally set

$$\delta\tau := \min\{\delta\tau_j : 1 \le j \le m\} \quad \text{and} \quad \Delta\tau := \max\{\delta\tau_j : 1 \le j \le m\}.$$

Throughout, we assume that $\Delta\tau/\delta\tau$ is bounded uniformly with respect to m.

Relation of Snapshot Grid Γ and Time Grid Λ^ℓ We desire to estimate the Galerkin error on the time grid $\Lambda^\ell = \{\tau_j\}_{j=0}^m$ by the POD error. Since this error depends on the snapshot grid $\Gamma = \{t_j\}_{j=0}^n$, we need to have a relation of the two grids:
For every $\tau_k \in \Lambda^\ell$, we wish to find an index $\overline{k}$ such that $t_{\overline{k}} \in \Gamma$ is closest to τ_k amongst all $t \in \Gamma$. In formal notation, this reads:

$$\overline{k} := \operatorname{argmin}\{\|\tau_k - t_j\| : 0 \le j \le n\}.$$

Furthermore, we need to ensure the right multiplicity of such estimations (especially in the case that the "ranges" of the grids do not match and the same $\overline{k}$ would appear several times). Thus we define $\sigma_n \in \{1, \ldots, n\}$ by

$$\sigma_n := \text{"maximum number of occurrences of the same value } t_{\overline{k}} \text{ as k ranges over } \{0, 1, \ldots, m\}\text{"}.$$

The Backward Euler Galerkin Problem According to Subsection 1.4.2, we may now introduce the fully discrete low-order model by approximating the time derivative by means of the implicit Euler scheme.

Problem 4.1.4 (Backward Euler Galerkin POD Problem)
Find a sequence $\{Y_k\}_{k=0}^m \subset \mathcal{V}^\ell$ satisfying

$$\left(\overline{\partial}_\tau Y_k, \psi\right)_H + a(Y_k, \psi) = (F(\tau_k), \psi)_H \quad \text{for all } \psi \in \mathcal{V}^\ell \text{ and } k = 1, \ldots, m, \tag{4.8a}$$

$$(Y_0, \psi)_H = (y_0, \psi)_H \quad \text{for all } \psi \in \mathcal{V}^\ell, \tag{4.8b}$$

where we have set:

$$\overline{\partial}_\tau Y_k := \frac{Y_k - Y_{k-1}}{\delta\tau_k}.$$

Existence of Solution For mathematical satisfaction we quote a result that a solution to this system actually exists and that there is some regularity information available.

Proposition 4.1.5 (Existence and a Priori Estimates for Solution)
For every $k = 1, \ldots, m$, there exists at least one solution Y_k of Problem 4.1.4. If $\Delta\tau$ is sufficiently small, the sequence $\{Y_k\}_{k=1}^m$ is uniquely determined. Moreover, there holds:

$$\|Y_k\|_H^2 \le (1 + \gamma\delta\tau)e^{-\gamma k\delta\tau} \|y_0\|_H^2 + \frac{1 - e^{-\gamma k\Delta\tau}}{\gamma} \|F\|_{C([0,T];H)}^2 \tag{4.9}$$

for $k = 0, \ldots, m$, where c_V, η and $\gamma = \eta / c_V^2$ are suitable constants. By means of these constants, we also obtain

$$\sum_{k=1}^{m} \|Y_k - Y_{k-1}\|_H^2 + \eta \sum_{k=1}^{m} \delta\tau_k \|Y_k\|_V^2 \leq \|y_0\|_H^2 + \frac{T}{\gamma} \|F\|_{C([0,T];H)}^2 . \tag{4.10}$$

Proof.
Refer to Kunisch and Volkwein 2002, Theorem 4.2, Appendix A and Kunisch and Volkwein 2001, Theorem 5. □

Discretization in Time and Solution Analogously to Subsection 1.4.2, we may solve the ODE system (4.7) for c^ℓ by the implicit Euler method, for instance. Furthermore, we could also build up the full linear system of solutions for each time step of Λ^ℓ (refer to (1.18)).

4.1.4 POD-ROM for FE Discretizations

So far, we have dealt with POD reduced-order models for "abstract" Evolution Problems. We now wish to focus on FE discretization of suitable Evolution Problems (such as parabolic IVPs).
In other words, as the Evolution of the previous subsection, we choose an FE system, i.e., we are now looking for a reduced approximation of an FE model. (Note that we may equally as well think of any Galerkin type approximation.)

Limitation of the Approach The approach proposed covers the practical calculation of a POD reduced-order modeling, yet its power in providing corresponding error estimates is limited (refer to Subsection 4.2.3).
In order to see this, let us reconsider the situation: In Chapter 3, we had already assumed, that the snapshots are known *exactly*, which in practice is of course not true – refer to the asymptotic analysis in terms of the snapshot grid with exact snapshots (Section 2.3) and the spatial approximation of the snapshots themselves (Subsection 3.2.4). If we now choose the FE system to be reduced by the POD Method, we "implicitly" assume it to be the *exact* model, whereas it actually is the "non-reduced" *discrete approximation* to an mathematically exact solution of an Evolution Problem.
In context of error estimates, this implies (even if we consider the snapshots to be known *exactly*): In the abstract setting, we may compare the "POD reduced" exact solution to the exact solution. In the practical context, we may compare the "POD reduced" FE solution to the "exact" FE solution. Yet we may not directly compare the "POD reduced" FE solution to the mathematically exact solution. This situation also is depicted in Figure 4.2 on page 79.

Coefficient Issue Technically, carrying out the POD Method on FE discretizations means carrying out the method on *coefficients* of FE basis functions. For example, we may represent the snapshots $\{\psi_j\}_{j=1}^{\ell}$ by means of a coefficient matrix $Y \in \mathbb{R}^{q \times n}$ in terms of the FE ansatz functions $\{\varphi_k\}_{k=1}^{q}$:

$$y_j = \sum_{i=1}^{q} Y_{ij} \varphi_i, \quad j = 1, \ldots, n. \tag{4.11}$$

Relation of Linear Systems Since we wish to reduce an "FE model", we would like to express the POD reduced-order system (4.7) in "terms" of the FE system (1.14), i.e., we wish to express the dependences of the respective matrices. For that purpose, we represent the POD Basis *in terms of the FE Basis* and obtain:

Proposition 4.1.6 (POD Projection)
Let $\{\varphi_i\}_{i=1}^q$ be FE ansatz functions and $\{\psi\}_{k=1}^\ell$ a POD Basis. Let $U \in \mathbb{R}^{q\times\ell}$ be the collection of coefficients of the POD Basis w.r.t. the FE ansatz functions:

$$\psi_i = \sum_{k=1}^{q} U_{ki}\varphi_k, \quad i = 1,\dots,\ell. \tag{4.12}$$

Then, for the matrices in the POD low-order system (4.7), we obtain

$$M^\ell = U^T M^T U, \qquad A^\ell = U^T A^T U, \qquad F^\ell = U^T \tilde{F}, \qquad g^\ell = U^T D\tilde{g}, \tag{4.13}$$

where M, A, $\tilde{F}$ and $\tilde{g}$ are taken from the FE system (refer to (1.13)). The FE vector $c \in C([0,T];\mathbb{R}^q)$ of the POD low-order solution y^ℓ is then given by

$$c(t) = Uc^\ell(t), \quad \text{i.e.,} \quad y^\ell(t) = \sum_{k=1}^{q} (Uc^\ell(t))^{(k)}\varphi_k, \quad y^\ell(0) = (U\alpha^\ell)^{(k)}\varphi_k.$$

Proof.
We essentially have to prove the relations of the matrices and then show the calculation of the low-order solution.

• Relations of Matrices We exemplary show the assertion $A^\ell = U^T A^T U$. All other cases might be obtained perfectly analogously. Note that for all entries of A^ℓ there holds

$$\begin{aligned} A^\ell_{ik} = a(\psi_k,\psi_i) &= a\Big(\sum_{j=1}^{q} U_{jk}\varphi_j, \sum_{l=1}^{q} U_{li}\varphi_l\Big) = \sum_{l,j=1}^{q} (U^T)_{kj} a(\varphi_j,\varphi_l) U_{li} \\ &= \sum_{l,j=1}^{q} (U^T)_{kj} A_{lj} U_{li} = \sum_{l,j=1}^{q} (U^T)_{kj}(A^T)_{jl} U_{li} = (U^T A^T U)_{ik}. \end{aligned}$$

• Low-order Solution Inserting the representation for the POD modes (4.12) into the definition of the low-order solution (4.5), we find

$$y^\ell(t) = \sum_{j=1}^{\ell} c_j^\ell(t)\psi_j = \sum_{j=1}^{\ell} c_j^\ell(t) \sum_{k=1}^{q} U_{kj}\varphi_k = \sum_{k=1}^{q}\sum_{j=1}^{\ell} U_{kj} c_j^\ell(t)\varphi_k = \sum_{k=1}^{q} (Uc^\ell(t))^{(k)}\varphi_k.$$

Analogously, we may proceed for the assertion on the initial value. □

4.2 Analysis of POD ROM – Error Estimates

After having set up the reduced-order model (Problem 4.1.4), we wish to establish respective error estimates.

Procedure Technically, we will use an (nearly arbitrary) grid for the snapshots and another one for the time integration with the implicit Euler method. The resulting error estimate will be improved such that convergence with decreasing time step size is ensured ("extension of snapshot set") and that the estimate is independent of the snapshot grid chosen ("asymptotic analysis").

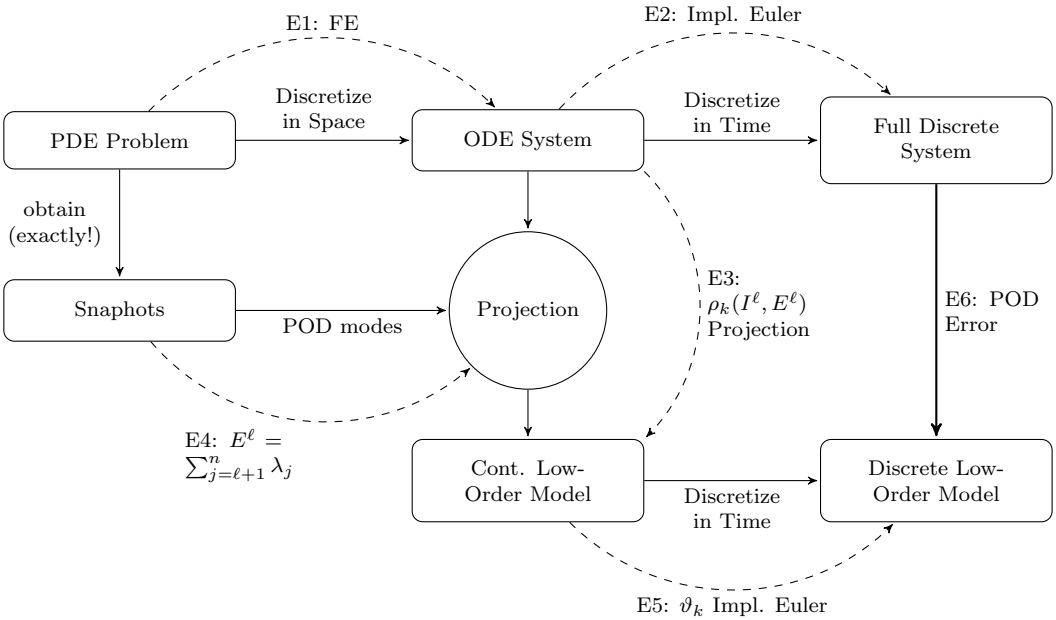

Figure 4.1: Sources of errors in a POD reduced-oder model

Throughout, we concentrate on the case $X = V$, yet we conclude by discussing the other "natural" choices of X in the POD Problem in this context: $X = H$. (Recall that for Evolution Problems we had setup a "Gelfand triple" $V \subset H = H^* \subset V^*$.) Furthermore, we confine ourselves to the case that the snapshots are known exactly. (Refer to Subsection 4.2.4 for a discussion of potential errors in the snapshots.)

Various Sources of Errors It should be obvious that an analysis of the POD reduced-order solution involves quite a variety of sources of errors, which are depicted in Figure 4.1 by dashed lines. We shall refer to these errors as we proceed in estimating the full error. In particular, the error in the low-order solution is due to the POD space error on the one hand and the error due to the time discretization on the other.

Exactness of Snapshots – Application to FE Models As mentioned above, we assume the snapshots to be known exactly. In context of reduced FE models (refer to Subsection 4.1.4), this assumption may be correct since the snapshots are given as "exact" FE coefficients (as in (4.11)). On the other hand, an actual interest in applying the POD Method is to use snapshots obtained from a *different* problem. Hence, there are "perturbation errors" in the snapshots even if they are known exactly. Therefore, this is an additional source of error which we will comment on in Subsection 4.2.4.

Restriction to Error Estimation for Discrete Case Reflecting on the previous section, it would be natural to establish error estimates for the (time-continuous) Galerkin POD Problem as well as another one for the (time-discretized) backward Euler Galerkin POD problem. But for the sake of brevity, we shall concentrate on the time-discrete case, i.e., on Problem 4.1.4. (Refer for example to Volkwein 2006, Theorem 2.1 for error estimates of a system continuous in time yet discrete in space.)

Assumptions on the Regularity of the Solution We aim to reduce the Evolution Problem 1.3.2 and for its solution $y \in W(0,T)$, we assume further:

(A1) $y_t \in L^2(0,T;V)$ and $y_{tt} \in L^2(0,T;H)$.

(A2) There exists a normed linear space $W \subset V$ (continuously embedded) and a constant $c_a > 0$ such that $y \in C([0,T];W)$ and

$$a(\phi,\psi) \leq c_a \|\phi\|_H \|\psi\|_W \quad \text{for all } \phi \in V \text{ and } \psi \in W. \tag{4.14}$$

Note that $V := H_0^1(\Omega)$ and $H := L^2(\Omega)$ (which are used in the application to parabolic IVP of second order) satisfy (A2).

Type of Estimates It shall turn out that we cannot rely on "typical approximation results" in function spaces. Hence, the error estimates are of an "unusual format". The estimates typically involve the time- and snapshot grid size and their relative position, the non-captured "energy" $\sum_{i=\ell+1}^{m} \lambda_i$ as well as the error in the projection of the initial value.

Preparations for Proof of Main Theorem Let us observe some properties of the POD Projection and establish an estimate of the projection of the initial value.

Lemma 4.2.1 (Ritz Projection and Norm Estimation)
Let $\{\psi_k\}_{k=1}^{\ell}$ be a POD Basis (obtained from snapshots which are assumed to be exact). The *Ritz Projection* $R^\ell : V \to \mathcal{V}^\ell$ for $1 \leq \ell \leq d$ and $\phi \in V$, which is characterized by

$$a(R^\ell\phi,\psi) = a(\phi,\psi) \quad \text{for all } \psi \in \mathcal{V}^\ell,$$

is given by the POD Projection P^ℓ of Definition 2.1.2. Furthermore, if (A2) holds, then there exists a constant $c_P(\ell,\lambda_\ell) > 0$ such that

$$\left\|P^\ell\right\|_{\mathcal{L}(V)} = 1 \quad \text{and} \quad \left\|P^\ell\right\|_{\mathcal{L}(H)} \leq c_P \quad \text{for } 1 \leq \ell \leq d. \tag{4.15}$$

Proof.
Since V is endowed with the inner product $a(\phi,\psi) := (\phi,\psi)_V$, and P^ℓ is an $(\cdot,\cdot)_{X=V}$-orthogonal projection of V onto $\mathcal{V}^\ell$, we have:

$$a(R^\ell\phi - \phi,\psi) = 0 \quad \text{for all } \psi \in \mathcal{V}^\ell,$$

which yields the assertion. P^ℓ being an $(\cdot,\cdot)_V$-orthogonal projection also yields $\left\|P^\ell\right\|_{\mathcal{L}(V)} = 1$. (For the assertion on $\left\|P^\ell\right\|_{\mathcal{L}(H)}$ refer to Kunisch and Volkwein 2002, Remark 4.4.) □

Lemma 4.2.2 (Initial Value Projection Estimation)
For a constant $C > 0$, there holds

$$\left\|y_0 - P^\ell y_0\right\|_H^2 \leq C \sum_{i=\ell+1}^{d(n)} (\psi_i, y_0)_V^2 .$$

Proof.
Due to the continuous embedding of V into H and Proposition 2.2.5, we obtain for $C := c_V^2 \max\{\alpha_i\}_{i=\ell+1}^{d(n)}$:

$$\left\|y_0 - P^\ell y_0\right\|_H^2 \leq c_V^2 \left\|y_0 - P^\ell y_0\right\|_V^2 \leq c_V^2 \sum_{i=\ell+1}^{d(n)} \alpha_i (\psi_i, y_0)_V^2 \leq C \sum_{i=\ell+1}^{d(n)} (\psi_i, y_0)_V^2 .$$

□

4.2.1 Basic Error Estimate

As mentioned above, we shall establish our *basic problem estimate* for two (nearly) independent time grids for the snapshots as well as the time integration. The space X may be thought of as a Hilbert space which shall be compatible with Problem 4.1.4 of concern, i.e., $X = V$ or $X = H$, where we concentrate on $X = V$ first. For simplicity of notation, define: $\|\cdot\| := \|\cdot\|_H$.

(Trapezoidal) Approximation of Error As we shall only be concerned about estimating problems discrete in time, we approximate the continuous error expression by a trapezoidal discretization:

$$\int_0^T \left\|P^\ell y(\tau) - y(\tau)\right\|_H^2 d\tau \;\approx\; \sum_{k=0}^{m} \beta_k \left\|Y_k - y(\tau_k)\right\|_H^2,$$

where $y(\tau)$ is the solution of the Evolution Problem 1.3.2 at time τ and $P^\ell y(\tau)$ denotes the POD Projection, i.e., the best approximation of $y(\tau)$ in $\mathcal{V}^\ell$. Thus, analogously to the weights α_j in the POD Problem, we choose the (positive) weights β_k to be:

$$\beta_0 := \frac{\delta\tau_1}{2}, \qquad \beta_j := \frac{\delta\tau_j + \delta\tau_{j+1}}{2} \quad \text{for } j = 1, \dots, m-1 \qquad \text{and} \qquad \beta_m = \frac{\delta\tau_m}{2}. \tag{4.16}$$

Decomposition of Error We aim to decompose the POD Galerkin error according to Figure 4.1. Note that ϑ_k denotes the error to the Projection of the *exact* solution.

Definition 4.2.3 (Decomposition of Error)
Decompose the POD Galerkin error expression

$$Y_k - y(\tau_k) = \vartheta_k + \rho_k$$

into the *time discretization error*

$$\vartheta_k := Y_k - P^\ell y(\tau_k)$$

as well as the *restriction to the POD subspace error*

$$\rho_k := P^\ell y(\tau_k) - y(\tau_k).$$

Treatment of the Time Discretization Error Let us start with estimating the error ϑ_k which is due to the time discretization. Note that we shall do so in two different ways, leading to:

Lemma 4.2.4 (Error Estimate for ϑ_k)
Assume that (A1) and (A2) hold and that $\Delta\tau$ is sufficiently small. Then, there exist constants $C_1, C_2 > 0$ independent of the grids Γ and Λ^ℓ such that

$$\begin{aligned}\sum_{k=0}^{m} \beta_k \left\|\vartheta_k\right\|_H^2 \le\; C_3\Big(\sum_{i=l+1}^{d(n)} (\psi_i, y_0)_V^2 + \frac{\sigma_n}{\delta t}\frac{1}{\delta\tau} \sum_{i=\ell+1}^{d(n)} \lambda_i \\ + \sigma_n(1 + c_P^2)\Delta\tau(\Delta\tau + \Delta t) \left\|y_{tt}\right\|_{L^2(0,T;V)}\Big),\end{aligned} \tag{4.17}$$

where $C_3 := C_1 T e^{C_2 T}$.

Alternatively, we may estimate (with a suitable constant C_3')

$$\sum_{k=0}^{m} \beta_k \left\|\vartheta_k\right\|^2 \leq C_3' \Big(\sum_{i=l+1}^{d(n)} (\psi_i, y_0)_V^2 + \sigma_n \Delta\tau \sum_{j=1}^{k} \sum_{i=\ell+1}^{d(n)} (y_t(t_j), \psi_i)_V^2 + \sigma_n (1 + c_P^2)(\Delta\tau^2 + \Delta\tau\Delta t) \left\|y_{tt}\right\|_{L^2(0,T;H)}^2 \Big).$$

Proof.
We transform Problem 4.1.4 into a Galerkin system for ϑ_k whose RHS we decompose into two terms z_k and w_k. We obtain an estimate for ϑ_k by testing in the Galerkin system with ϑ_k as well. We estimate w_k and decompose z_k into two terms D and Q. We estimate D and treat Q in two different ways. Combining all these findings, we arrive at both the assertions.

• Galerkin System for ϑ_k Define:

$$\overline{\partial}_\tau \vartheta_k := \frac{\vartheta_k - \vartheta_{k-1}}{\delta\tau_k} \quad \text{for } k = 1, \ldots, m. \tag{4.18}$$

By inserting $Y_k = \vartheta_k + P^\ell y(\tau_k)$ (refer to the definition of ϑ_k) into the model equation (4.8a), we obtain (by linearity) for all $\psi \in \mathcal{V}^\ell$ and $k = 1, \ldots, m$:

$$\left(\overline{\partial}_\tau \vartheta_k, \psi\right)_H + \left(\overline{\partial}_\tau P^\ell y(\tau_k), \psi\right)_H + a(\vartheta_k, \psi) + a(P^\ell y(\tau_k), \psi) = (F(\tau_k), \psi)_H . \tag{4.19}$$

We define

$$v_k := y_t(\tau_k) - \overline{\partial}_\tau P^\ell y(\tau_k)$$

and rearrange (4.19) as follows (again for all $\psi \in \mathcal{V}^\ell$):

$$\begin{aligned} \left(\overline{\partial}_\tau \vartheta_k, \psi\right)_H + a(\vartheta_k, \psi) &= (F(\tau_k), \psi)_H - a(P^\ell y(\tau_k), \psi) - \left(\overline{\partial}_\tau P^\ell y(\tau_k), \psi\right)_H \\ &= (y_t(\tau_k), \psi)_H - \left(\overline{\partial}_\tau P^\ell y(\tau_k), \psi\right)_H \\ &= (v_k, \psi)_H , \end{aligned} \tag{4.20}$$

where the second step is established by making use of the *full*-order model. In particular due to (4.1a), we find:

$$\frac{d}{dt} (y(t), \varphi)_H = (F(t), \varphi)_H - a(y(t), \varphi) \quad \text{for all } \varphi \in V, \quad t \in (0, T].$$

Due to $\mathcal{V}^\ell \subset X := V$ and Lemma 4.2.1, we then have (as used in (4.20)):

$$(y_t(t), \varphi)_H = (F(t), \varphi)_H - a(P^\ell y(t), \varphi) \quad \text{for all } \varphi \in \mathcal{V}^\ell, \quad t \in (0, T].$$

• Decomposition of RHS We decompose $v_k = w_k + z_k$, where

$$w_k := y_t(\tau_k) - \overline{\partial}_\tau y(\tau_k) \quad \text{and} \quad z_k := \overline{\partial}_\tau y(\tau_k) - \overline{\partial}_\tau P^\ell y(\tau_k). \tag{4.21}$$

• Testing with ϑ_k We now chose $\psi := \vartheta_k \in \mathcal{V}^\ell$ as test functions in (4.20):

$$\left(\overline{\partial}_\tau \vartheta_k, \vartheta_k\right)_H + a(\vartheta_k, \vartheta_k) = (v_k, \vartheta_k)_H .$$

Using (4.18) and multiplying by $2\delta\tau_k$, we obtain (due to ellipticity of a in (A2))

$$(\vartheta_k - \vartheta_{k-1}, \vartheta_k)_H + 2\delta\tau_k\eta \|\vartheta_k\|^2 \leq 2\delta\tau_k (v_k, \vartheta_k)_H .$$

Using the Schwarz inequality, we get:

$$\|\vartheta_k\|^2 - \|\vartheta_{k-1}\|^2 + \|\vartheta_k - \vartheta_{k-1}\| + 2\delta\tau_k\eta \|\vartheta_k\|^2 \leq 2\delta\tau_k \|v_k\| \|\vartheta_k\| .$$

Omitting the positive terms on the left, we infer

$$\|\vartheta_k\|^2 \leq \|\vartheta_{k-1}\|^2 + 2\delta\tau_k \|v_k\| \|\vartheta_k\| .$$

• Estimate; Applying Decomposition We may now obtain an estimate for $\|\vartheta_k\|$, using Young's inequality and gathering terms appropriately:

$$\begin{aligned}
\|\vartheta_k\|^2 &\leq \|\vartheta_{k-1}\|^2 + 2\delta\tau_k \|w_k + z_k\| \|\vartheta_k\| \\
&\leq \|\vartheta_{k-1}\|^2 + 2\delta\tau_k (\|w_k\| \|\vartheta_k\| + \|z_k\| \|\vartheta_k\|) \\
&\leq \|\vartheta_{k-1}\|^2 + 2\delta\tau_k \left(1/2 \|w_k\|^2 + 1/2 \|\vartheta_k\|^2 + 1/2 \|z_k\|^2 + 1/2 \|\vartheta_k\|^2\right) \\
&\leq \|\vartheta_{k-1}\|^2 + \delta\tau_k \left(\|w_k\|^2 + \|z_k\|^2 + 2 \|\vartheta_k\|^2\right),
\end{aligned}$$

which we may solve for $\|\vartheta_k\|^2$:

$$(1 - 2\delta\tau_k) \|\vartheta_k\|^2 \leq \left(\|\vartheta_{k-1}\|^2 + \delta\tau_k \left(\|w_k\|^2 + \|z_k\|^2\right)\right). \tag{4.22}$$

• Summing with Factor Estimation Now suppose $\Delta\tau \leq \frac{1}{4}$. Then: $1-2\delta\tau_k \geq 1-2\Delta\tau \geq \frac{1}{2}$, which yields

$$\frac{1}{1 - 2\delta\tau_k} \leq \frac{1}{1 - 2\Delta\tau} = \frac{1 - 2\Delta\tau + 2\Delta\tau}{1 - 2\Delta\tau} = 1 + \frac{2\Delta\tau}{1 - 2\Delta\tau} \leq 1 + 4\Delta\tau. \tag{4.23}$$

In the following step, we will make use of the observation ($a_k, b_k \in \mathbb{R}, q > 0$):

$$a_k \leq q \cdot (a_{k-1} + b_k) \quad k = 1, \ldots, n \quad \text{implies} \quad a_k \leq a_0 q^k + \sum_{j=0}^{k} b_j$$

Using this, from (4.22) and (4.23), we obtain by summing on k

$$\begin{aligned}
\|\vartheta_k\|^2 &\leq (1 + 4\Delta\tau) \left(\|\vartheta_{k-1}\|^2 + \delta\tau_k \left(\|w_k\|^2 + \|z_k\|^2\right)\right) \\
&\leq \left(1 + 4\Delta\tau \frac{\delta\tau k}{\delta\tau} \frac{1}{k}\right)^k \left(\|\vartheta_0\|^2 + \sum_{j=1}^{k} \delta\tau_j \left(\|w_j\|^2 + \|z_j\|^2\right)\right) \\
&\leq e^{ckT} \left(\|\vartheta_0\|^2 + \sum_{j=1}^{k} \delta\tau_j \left(\|w_j\|^2 + \|z_j\|^2\right)\right).
\end{aligned} \tag{4.24}$$

The last estimation is due to: $(1 + a/n)^n$ approaches e^a from below (for $a \in \mathbb{R}$), i.e.,

$$\left(1 + \frac{4\Delta\tau}{\delta\tau} \frac{k\delta\tau}{k}\right)^k \leq e^{ck\delta\tau} \leq e^{ckT} \quad \text{with } c := 4\frac{\Delta\tau}{\delta\tau} \text{ (bounded by assumption)}$$

• Estimation of w_k-Term According to Kunisch and Volkwein 2002, (B.15) we may estimate the w_k-Term as follows:

$$\sum_{j=1}^{k} \delta\tau_j \left\|w_j\right\|^2 \leq \frac{\Delta\tau^2}{3} \left\|y_{tt}\right\|^2_{L^2(0,\tau_k;H)}. \tag{4.25}$$

• Estimation of z_k-Term We are interested in estimating (see definition in (4.21))

$$z_k = \overline{\partial}_\tau y(\tau_j) - \overline{\partial}_\tau P^\ell y(\tau_j),$$

given on the time grid Λ^ℓ. In order to apply the theory on POD with $X = V$, we need to estimate z_k by an expression depending on the snapshot grid Γ. We manage this by "zero-adding" the terms $y_t(\tau_j)$, $y_t(t_{\bar{j}})$, $\overline{\partial}_\tau y(t_{\bar{j}})$ as well as their respective images under P^ℓ. Using the triangular inequality, we obtain the estimation

$$\left\|z_k\right\|^2 \leq D + 7 \underbrace{\left\|\overline{\partial}_\tau y(t_{\bar{j}}) - \overline{\partial}_\tau P^\ell y(t_{\bar{j}})\right\|^2}_{:=Q}, \tag{4.26}$$

D denotes a term $D\big(\overline{\partial}_\tau y(\tau_j), y_t(\tau_j), y_t(t_{\bar{j}}), \overline{\partial}_\tau y(t_{\bar{j}})\big)$, whose explicit form is not of importance, but which might be estimated in the fashion of (4.25) to yield

$$\begin{aligned} D \leq \frac{7}{3}(1+c_P^2)\delta\tau_j \big(\left\|y_{tt}\right\|^2_{L^2(\tau_{j-1},\tau_j;H)} + \left\|y_{tt}\right\|^2_{L^2(t_{\bar{j}-1},t_{\bar{j}};H)}\big) \\ + 14(1+c_P^2)\Delta t \left\|y_{tt}\right\|^2_{L^2(t_{\bar{j}-1},t_{\bar{j}+1};H)}. \end{aligned} \tag{4.27}$$

For all the respective details refer to the proof of Kunisch and Volkwein 2002, (B.16). Summing this estimation over j, we may estimate

$$\sum_{j=1}^{k} \delta\tau_j D \leq 14\sigma_n(1+c_P^2)(\Delta\tau^2 + \Delta\tau\Delta t) \left\|y_{tt}\right\|^2_{L^2(0,t_{\bar{k}+1};H)}. \tag{4.28}$$

It is now that we are left with treating Q. We may do so in two different ways in order to yield the assertions of the lemma.

• First Variant for Q Using the definition of $\overline{\partial}_\tau$ and triangular inequality, we rearrange

$$\begin{aligned} \delta\tau_j Q &= \delta\tau_j \left\|\overline{\partial}_\tau y(t_{\bar{j}}) - \overline{\partial}_\tau P^\ell y(t_{\bar{j}})\right\|^2 \\ &\leq \frac{2}{\delta\tau_j}\big(\left\|y(t_{\bar{j}}) - P^\ell y(t_{\bar{j}})\right\|^2 + \left\|y(t_{\bar{j}-1}) - P^\ell y(t_{\bar{j}-1})\right\|^2\big). \end{aligned} \tag{4.29}$$

According to our final usage of the estimation of z_k in (4.24), we have shifted $\delta\tau_j$ to the left. Note that we see the factor $\frac{1}{\delta\tau_j}$ appear on the RHS, which shall be a matter of improvement in the subsection to come.
We may estimate the sum over the first additive term over j by using $\alpha_j \geq \delta t/2$ (i.e., $2\alpha_j/\delta t \geq 1$) and the continuous injection of V into H in order to use the POD error estimate for $X = V$:

$$\sum_{j=1}^{k} \frac{1}{\delta\tau_j} \left\|y(t_{\bar{j}}) - P^\ell y(t_{\bar{j}})\right\|^2 \leq \frac{2\sigma_n}{\delta t\delta\tau} \sum_{j=1}^{k} \alpha_j \left\|y(t_j) - P^\ell y(t_j)\right\|^2 \leq \frac{2\sigma_n c_V^2}{\delta t\delta\tau} \sum_{i=\ell+1}^{d(n)} \lambda_i. \tag{4.30}$$

Since the additive terms in (4.29) only differ in an index, we may estimate the second term in the very same way to find

$$\sum_{j=1}^{k} \delta\tau_j Q \leq \frac{8\sigma_n c_V^2}{\delta t \delta\tau} \sum_{i=\ell+1}^{d(n)} \lambda_i.$$

Using this estimate together with estimate (4.28) for D, we due to (4.26) arrive at

$$\sum_{j=1}^{k} \delta\tau_j \left\|z_j\right\|^2 \leq 14\sigma_n(1+c_P^2)(\Delta\tau^2+\Delta\tau\Delta t)\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}+1};H)} + \frac{56\sigma_n c_V^2}{\delta t\delta\tau}\sum_{i=\ell+1}^{d(n)}\lambda_i.$$

• Obtaining the First Assertion Inserting the last estimation as well as the estimation (4.25) for w_k, we infer from (4.24)

$$\begin{aligned}\left\|\vartheta_k\right\|^2 \leq e^{ckT}\Big(&\left\|\vartheta_0\right\|^2 + \frac{\Delta\tau^2}{3}\left\|y_{tt}\right\|^2_{L^2(0,\tau_k;H)}\\ &+ 14\sigma_n(1+c_P^2)(\Delta\tau^2+\Delta\tau\Delta t)\left\|y_{tt}\right\|_{L^2(0,t_{\overline{k}+1};H)} + \frac{56\sigma_n c_V^2}{\delta t\delta\tau}\sum_{i=\ell+1}^{d(n)}\lambda_i\Big),\end{aligned}$$

which we might transform to (for each $1 \leq k \leq m$ and suitable constants C_1, C_2)

$$\begin{aligned}\left\|\vartheta_k\right\|_H^2 \leq C_1 e^{C_2 kT}\Big(&\left\|y_0 - P^\ell y_0\right\|_H^2 + \frac{\sigma_n}{\delta t}\frac{1}{\delta\tau}\sum_{i=\ell+1}^{d(n)}\lambda_i\\ &+ \sigma_n(1+c_p^2)\Delta\tau(\Delta\tau+\Delta t)\left\|y_{tt}\right\|_{L^2(0,t_{\overline{k}+1};H)}\Big).\end{aligned} \tag{4.31}$$

Summing on k and using Lemma 4.2.2, this yields the first assertion of the lemma.

• Second Variant for Q Instead of rearranging as in (4.29), we directly "zero-add" the term $y_t(t_{\overline{j}})$ as well as its projection. Then, by manipulations similar to Volkwein 2006, p. 26 and minding the norm estimation of the projection of Lemma 4.2.1, we obtain

$$Q = \left\|\overline{\partial}_\tau y(t_{\overline{j}}) - \overline{\partial}_\tau P^\ell y(t_{\overline{j}})\right\|^2 \leq (4+2c_P^2)\left\|y_t(t_{\overline{j}}) - \overline{\partial}_\tau y(t_{\overline{j}})\right\|^2 + 4\left\|y_t(t_{\overline{j}}) - P^\ell y_t(t_{\overline{j}})\right\|^2.$$

The first additive term, we estimate analogously to (4.25) ("estimation for w_k"). The second term, we may treat similarly to Lemma 4.2.2 (according to (A1), there holds $y_t(t) \in V, t \in [0,T]$):

$$\left\|y_t(t_{\overline{j}}) - P^\ell y_t(t_{\overline{j}})\right\|^2 \leq \sigma_n\left\|y_t(t_j) - P^\ell y_t(t_j)\right\|^2 \leq C\sigma_n c_V^2 \sum_{i=\ell+1}^{d(n)} \left(y_t(t_j),\psi_i\right)_V^2,$$

where we have set $C := \max\{\alpha_i\}_{i=\ell+1}^{d(n)}$. Altogether, we arrive at the estimation

$$Q \leq 2(2+c_P^2)\frac{\delta\tau_j}{3}\left\|y_{tt}\right\|^2_{L^2(t_{\overline{j}-1},t_{\overline{j}};H)} + 4C\sigma_n c_V^2\sum_{i=\ell+1}^{d(n)}\left(y_t(t_j),\psi_i\right)_V^2,$$

which yields for the "sum of interest" (using $\delta\tau_j \leq \Delta\tau$):

$$\sum_{j=1}^{k}\delta\tau_j Q \leq 2(2+c_P^2)\frac{\Delta\tau^2}{3}\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}};H)} + 4C\sigma_n c_V^2\Delta\tau\sum_{j=1}^{k}\sum_{i=\ell+1}^{d(n)}\left(y_t(t_j),\psi_i\right)_V^2.$$

Using this estimate together with estimate (4.28) for D, we due to (4.26) arrive at

$$\sum_{j=1}^{k} \delta\tau_j \left\|z_j\right\|^2 \leq 14\sigma_n(1+c_P^2)(\Delta\tau^2+\Delta\tau\Delta t)\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}+1};H)} + 14(2+c_P^2)\frac{\Delta\tau^2}{3}\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}};H)} + 28C\sigma_n c_V^2\Delta\tau\sum_{j=1}^{k}\sum_{i=\ell+1}^{d(n)}(y_t(t_j),\psi_i)_V^2 .$$

• Obtaining the Second Assertion Inserting the last estimation as well as the estimation (4.25) for w_k into (4.24), we infer (for each k):

$$\begin{aligned}\left\|\vartheta_k\right\|^2 \leq e^{ck\delta\tau}\Big(&\left\|\vartheta_0\right\|^2 + \frac{\Delta\tau^2}{3}\left\|y_{tt}\right\|^2_{L^2(0,\tau_k;H)} \\ &+ 14\sigma_n(1+c_P^2)(\Delta\tau^2+\Delta\tau\Delta t)\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}+1};H)} \\ &+ 14(2+c_P^2)\frac{\Delta\tau^2}{3}\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}};H)} + 28C\sigma_n c_V^2\Delta\tau\sum_{j=1}^{k}\sum_{i=\ell+1}^{d(n)}(y_t(t_j),\psi_i)_V^2 ,\end{aligned}$$

which we may summarize (introducing respective constants C_1' and C_2') to

$$\begin{aligned}\left\|\vartheta_k\right\|^2 \leq C_1' e^{C_2' k\delta\tau}\Big(&\left\|y_0 - P^\ell y_0\right\|_H^2 + \sigma_n\Delta\tau\sum_{j=1}^{k}\sum_{i=\ell+1}^{d(n)}(y_t(t_j),\psi_i)_V^2 \\ &+ \sigma_n(1+c_P^2)(\Delta\tau^2+\Delta\tau\Delta t)\left\|y_{tt}\right\|^2_{L^2(0,t_{\overline{k}+1};H)}\Big).\end{aligned}$$

Summing on k and using Lemma 4.2.2, the second assertion of the lemma follows. □

Estimation of the POD Projection Contribution Let us now take care of the second contribution to the POD Galerkin error: the error ρ_k due to the POD subspace restriction.

Lemma 4.2.5 (Error Estimate for $\boldsymbol{\rho_k}$)
Assume that (A1) and (A2) hold and that $\Delta\tau$ is sufficiently small. Then, there exists a $C_4 > 0$ independent of the grids Γ and Λ^ℓ such that

$$\sum_{k=0}^{m}\beta_k\left\|\rho_k\right\|_H^2 \leq C_4\Big(\sigma_n(1+c_P^2)\Delta\tau\Delta t\left\|y_t\right\|_{L^2(0,T;H)} + \frac{\sigma_n\Delta\tau}{\delta t}\sum_{i=\ell+1}^{d(n)}\lambda_i\Big). \tag{4.32}$$

Proof.
We wish to estimate the error on the time grid Λ^ℓ whereas the POD error estimation is established on the snapshot grid Γ. Thus, we need to bring together the two grids. We shall therefore obtain an additive term which is due to the "connection" of the grids and one term which is obtained from the actual POD error estimate.

• Decomposition into two Contributions Using Young's inequality and noting that for

$a, b, c \in \mathbb{R}$ there holds $(a+b+c)^2 \leq 3(a^2+b^2+c^2)$, we obtain

$$\begin{aligned}\|\rho_k\|^2 &= \left\|P^\ell y(\tau_k) - y(\tau_k)\right\|^2 \\ &\leq 3\left(\left\|P^\ell y(\tau_k) - P^\ell y(t_{\overline{k}})\right\|^2 + \left\|P^\ell y(t_{\overline{k}}) - y(t_{\overline{k}})\right\|^2 + \left\|y(t_{\overline{k}}) - y(\tau_k)\right\|^2\right) \\ &\leq 3(1+c_P^2)\left\|y(t_{\overline{k}}) - y(\tau_k)\right\|^2 + 3\left\|P^\ell y(t_{\overline{k}}) - y(t_{\overline{k}})\right\|^2,\end{aligned} \tag{4.33}$$

where the last inequality is due to the boundedness of P^ℓ in $\|\cdot\| \equiv \|\cdot\|_H$ (Lemma 4.2.1) and:

$$\left\|P^\ell\left(y(\tau_k) - y(t_{\overline{k}})\right)\right\|^2 \leq \|P\|^2_{\mathcal{L}(H)}\left\|y(\tau_k) - y(t_{\overline{k}}\right\|^2.$$

• Estimate for Time Grid Contribution Using the triangular inequality, we get

$$\begin{aligned}\left\|y(t_{\overline{k}}) - y(\tau_k)\right\|^2 &\leq \left(\int_{\tau_k}^{t_{\overline{k}}} \|y_t(s)\| \,\mathrm{d}s\right)^2 \\ &\leq \left(\int_{t_{\overline{k}-1}}^{t_{\overline{k}+1}} \|y_t(s)\| \,\mathrm{d}s\right)^2 \\ &\leq (\delta t_{\overline{k}} + \delta t_{\overline{k}+1})\,\|y_t\|_{L^2(t_{\overline{k}-1}, t_{\overline{k}+1}; H)},\end{aligned} \tag{4.34}$$

where we set $t_{m+1} := T$ whenever $\overline{k} = m$.

• Summation for the Time Grid Contribution We wish to show that (4.34) implies

$$\sum_{k=0}^{m} \beta_k \left\|y(t_{\overline{k}}) - y(\tau_k)\right\|^2 \leq 2\sigma_n \Delta\tau\Delta t\,\|y_t\|^2_{L^2(0,T;H)}. \tag{4.35}$$

By definition of β_k, we have $\beta_k \leq \Delta\tau$. Furthermore, we may estimate $\delta t_{\overline{k}} + \delta t_{\overline{k}+1} \leq 2\Delta t$ as (by definition) $\Delta t \geq \delta t_k$ for all $k = 1, \ldots, n$.
Hence, the following estimate yields the assertion:

$$\begin{aligned}\sum_{k=0}^{m} \|y_t\|^2_{L^2(t_{\overline{k}-1}, t_{\overline{k}+1}; H)} &= \sum_{k=0}^{m} \int_{t_{\overline{k}-1}}^{t_{\overline{k}+1}} \|y_t(s)\| \,\mathrm{d}s \\ &\leq \sum_{k=0}^{m} \sigma_n \int_{t_{k-1}}^{t_{k+1}} \|y_t(s)\| \,\mathrm{d}s \\ &\leq \sigma_n \int_0^T \|y_t(s)\| \,\mathrm{d}t = \sigma_n \|y_t\|^2_{L^2(0,T;H)}.\end{aligned}$$

• Summation for the POD Error Estimation In the following estimation, we use that V is continuously embedded in H with constant c_V and that $\beta_k \leq \Delta\tau$ by definition. We then estimate the sum over those $t_{\overline{k}}$ which are closest to some τ_k by the sum over all t_k (taking care of possible multiplicities by σ_n).
Since $\alpha_j \geq \delta t/2$ (i.e. $2\alpha_j/\delta t \geq 1$), we may expand the estimation by this term in order

to apply the POD error estimation of Corollary 2.3.4:

$$\begin{aligned}\sum_{k=0}^{m}\beta_k\left\|P^\ell y(t_{\overline{k}})-y(t_{\overline{k}})\right\|_H^2 &\le c_V^2\Delta\tau\sum_{k=0}^{m}\left\|P^\ell y(t_{\overline{k}})-y(t_{\overline{k}})\right\|_V^2\\ &\le c_V^2\Delta\tau\sigma_n\sum_{j=0}^{n}\left\|P^\ell y(t_j)-y(t_j)\right\|_V^2\\ &= c_V^2\Delta\tau\sigma_n\sum_{j=0}^{n}\frac{2\alpha_j}{\delta t}\left\|P^\ell y(t_j)-y(t_j)\right\|_V^2\\ &= \frac{2c_V^2\sigma_n\Delta\tau}{\delta t}\sum_{i=\ell+1}^{d(n)}\lambda_i.\end{aligned}$$

• Obtaining the Assertion Summing on k in (4.33) and using the last estimate as well as (4.35), we obtain

$$\sum_{k=0}^{m}\beta_k\left\|\rho_k\right\|_H^2 \le 6\sigma_n(1+c_P^2)\Delta\tau\Delta t\left\|y_t\right\|_{L^2(0,T;H)}+\frac{6c_V^2\sigma_n\Delta\tau}{\delta t}\sum_{i=l+1}^{d(n)}\lambda_i, \tag{4.36}$$

which (apart from the introduction of C_1) equals the assertion. □

Actual Error Estimate Let us now combine the two lemmas above to the actual error estimate of desire:

Theorem 4.2.6 (Error Estimate)
Assume that (A1) and (A2) hold and that $\Delta\tau$ is sufficiently small. Then, there exists a constant $C(T)$, independent of the grids Γ and Λ^ℓ, such that

$$\begin{aligned}\sum_{k=0}^{m}\beta_k\left\|Y_k-y(\tau_k)\right\|_H^2 \le\ & C\sum_{i=\ell+1}^{d(n)}\left((\psi_i,y_0)_V^2+\frac{\sigma_n}{\delta t}\Big(\frac{1}{\delta\tau}+\Delta\tau\Big)\lambda_i\right)\\ &+C\sigma_n(1+c_P^2)\Delta\tau\left(\Delta t\left\|y_t\right\|_{L^2(0,T;H)}^2+(\Delta\tau+\Delta t)\left\|y_{tt}\right\|_{L^2(0,T;V)}^2\right).\end{aligned}$$

Alternatively, we may estimate for a constant $C_2(T)$:

$$\begin{aligned}\sum_{k=0}^{m}\beta_k\left\|Y_k-y(\tau_k)\right\|_H^2 \le\ & C_2\sum_{i=\ell+1}^{d(n)}\left((\psi_i,y_0)_V^2+\frac{\sigma_n}{\delta t}\Delta\tau\lambda_i+\sigma_n\Delta\tau\sum_{j=1}^{k}(y_t(t_j),\psi_i)_V^2\right)\\ &+C_2\sigma_n(1+c_P^2)\Delta\tau\left(\Delta t\left\|y_t\right\|_{L^2(0,T;H)}^2+(\Delta\tau+\Delta t)\left\|y_{tt}\right\|_{L^2(0,T;V)}^2\right).\end{aligned}$$

Proof.
Since the two assertion differ only slightly, let us establish them simultaneously (by means of two choices for the term Q). By Definition 4.2.3 of ϑ_k and ρ_k, we obtain from Lemmas 4.2.4 and 4.2.5 (using Young's inequality in the first estimation and defining

$\overline{C}_3 := \max\{C_3, C_3'\}$:

$$\begin{aligned}\sum_{k=0}^{m} \beta_k \left\|Y_k - y(\tau_k)\right\|_H^2 &= \sum_{k=0}^{m} \beta_k \left\|\vartheta_k + \rho_k\right\|_H^2 \leq 2\sum_{k=0}^{m} \beta_k \left\|\vartheta_k\right\|_H^2 + 2\sum_{k=0}^{m} \beta_k \left\|\rho_k)\right\|_H^2 \\ &\leq 2\overline{C}_3\Big(\sum_{i=\ell+1}^{d(n)} (\psi_i, y_0)_V^2 + Q\Big) \\ &\quad + \sigma_n(1+c_P^2)\Delta\tau(\Delta\tau+\Delta t)\left\|y_{tt}\right\|_{L^2(0,T;V)}\Big) \\ &\quad + 2C_4\Big(\sigma_n(1+c_P^2)\Delta\tau\Delta t\left\|y_t\right\|_{L^2(0,T;H)} + \frac{\sigma_n\Delta\tau}{\delta t}\sum_{i=\ell+1}^{d(n)}\lambda_i\Big),\end{aligned}$$

where, depending on the alternative in Lemma 4.2.4, $Q \in \{Q_1, Q_2\}$ with

$$Q_1 := \frac{\sigma_n}{\delta t}\frac{1}{\delta\tau}\sum_{i=\ell+1}^{d(n)}\lambda_i \quad \text{and} \quad Q_2 := \sigma_n\Delta\tau\sum_{j=1}^{k}\sum_{i=\ell+1}^{d(n)}(y_t(t_j), \psi_i)_V^2\,.$$

For $Q = Q_1$, we additionally observe that (with a constant $\overline{C} > 0$) we may summarize:

$$\overline{C}_3 Q_1 + \frac{12c_V^2\sigma_n\Delta\tau}{\delta t}\sum_{i=l+1}^{d(n)}\lambda_i = \overline{C}\frac{\sigma_n}{\delta t}\left(\frac{1}{\delta\tau}+\Delta\tau\right)\sum_{i=l+1}^{d(n)}\lambda_i.$$

After some reordering of (sum-) terms, we may choose $C_1 > 0$ and $C_2 > 0$ suitably such that both the assertions of Theorem 4.2.6 follow. □

4.2.2 Improvements of the Basic Error Estimate

We improve the error estimations of Theorem 4.2.6 in three ways: We extend the snapshot set in order to prevent the factor $\frac{1}{\delta\tau}$ or the non-modeled derivative (depending on the choice of estimate). In particular, we find that in this setting both the approaches coincide (up to a constant). Furthermore, assuming some additional regularity on the solution of the Evolution System as well as the time grids, shall tidy up the expression. Finally, the asymptotic analysis in the snapshot grid (with exact snapshots) of Section 2.3 shall help us to omit the dependency of the error estimates on the snapshot grid.

Extension of the Snapshot Set In the first estimation of Theorem 4.2.6, one additive term depends on $\frac{1}{\delta\tau}$. From a theoretical viewpoint, this is not desired as we wish the discrete solution to convergence to the exact one by decreasing $\Delta\tau$ (and hence decreasing $\delta\tau$).

From a numerical point of view, $\sum_{i=\ell+1}^{d(n)}\lambda_i$ generally is small in comparison to $\Delta\tau$. Yet in Hömberg and Volkwein 2003, Subsection 3.4.2 it was shown that this approach also improves the numerical results – although one has to keep in mind that the eigenvalue problem to solve in order to obtain the POD Basis nearly doubles in size.

One way of overcoming this problem is to consider the alternative estimation: The additive term depending on $\frac{1}{\delta\tau}$ is *replaced* by the "non-modeled derivative" of the solution. This contribution is not desired either, yet we may let it vanish by "extending" the snapshot set: Let us denote the respective POD Basis by $\{\hat{\psi}_i\}_{i=1}^{\ell}$ and their "energy" contributions by $\{\hat{\lambda}_i\}_{i=1}^{\ell}$. We extend the canonical snapshot set for the POD Method by the finite differences of the snapshots (as in (3.6)).

By including the difference quotients into the snapshot set (refer to the discussion on setting up the snapshot set in Subsection 3.2.1), we include the previously "non-modeled" derivative into the POD

Basis. Hence, we may estimate the term by the non-modeled energy of the extended snapshot set: $\{\hat{\lambda}_i\}_{i=\ell+1}^{d(n)}$. Since a term of this type is already present in the estimate, this additional term simply changes the constant of the estimation (and in this sense "vanishes").

Approaches Coincide In Kunisch and Volkwein 2002, Corollary 4.13 it was shown that for an "extended" snapshot set the unwanted additive term in the first assertion of Theorem 4.2.6 may also be estimated by a term depending on $\{\hat{\lambda}_i\}_{i=\ell+1}^{d(n)}$. Therefore, for an extended snapshot set, the two approaches for the error estimation coincide (up to a constant).

More Regularity in Solution Let $y \in W^{2,2}(0,T;V)$ hold. As there now holds $y_{tt}(t) \in V$, we may estimate all H-dependent norms by respective V-norms (due to the continuous embedding with constant c_V). We may then estimate the projection in the $\mathcal{L}(V)$-norm instead of the $\mathcal{L}(H)$-norm. Thus, instead of c_P we obtain c_V, which might be hidden in a general constant. (This was not possible for c_P since it depended on the POD Projection, which in turn depended on the POD Basis, obtained for a specific snapshot grid Γ_n.)

Additional Assumptions on Time Grids Assume that there is not "too much variance" in the time grids:

$$\Delta t = O(\delta\tau) \qquad \text{and} \qquad \Delta\tau = O(\delta t).$$

Then, there exists a constant $C(T)$, independent of ℓ and the grids Γ and Λ^ℓ, such that

$$\max(\sigma_n, \frac{\sigma_n \Delta t}{\delta t}) \leq C(T). \tag{4.37}$$

Snapshot Grid Independent Version Note that the error estimate of Theorem 4.2.6 depends on the snapshot grid Γ since the λ_i do. Thus, we shall make use of an "ideal" POD Basis $\{\hat{\psi}^\infty\}_{i=1}^{\ell}$ (refer to the asymptotic analysis in Section 2.3) in order to estimate the error by an expression independent of the snapshot grid.

Simplified and Improved Error Estimate Let us summarize these findings in the following corollary, which actually is Kunisch and Volkwein 2002, Corollary 4.13. Note that this actually presents a simplification of both the estimates of Theorem 4.2.6.

Corollary 4.2.7 (Asymptotic Estimate)
Assume $y \in W^{2,2}(0,T;V)$. Setup an "extended" snapshot set $\hat{\mathcal{V}}$ as in (3.6) (by including the difference quotients of the snapshots). Suppose

$$\Delta t = O(\delta\tau) \qquad \text{and} \qquad \Delta\tau = O(\delta t)$$

and choose $\ell \in \mathbb{N}$ such that

$$\hat{\lambda}_\ell^\infty \neq \hat{\lambda}_\ell^\infty.$$

Then, there exist a constant $C(T)$, independent of ℓ and the grids Γ and Λ^ℓ, as well as $\overline{\Delta t} > 0$, depending on ℓ, such that for all $\Delta t \leq \overline{\Delta t}$ there holds

$$\begin{aligned}\sum_{k=0}^{m} \beta_k \left\| Y_k - y(\tau_k) \right\|_H^2 \leq\ & C \sum_{i=\ell+1}^{\infty} \left(\hat{\psi}_i^\infty, y_0 \right)_V + C \sum_{i=\ell+1}^{\infty} \hat{\lambda}_i^\infty \\ & + C\Delta\tau\Delta t \left\| y_t \right\|_{L^2(0,T;V)}^2 + C\Delta\tau(\Delta\tau + \Delta t) \left\| y_{tt} \right\|_{L^2(0,T;V)}^2 .\end{aligned} \tag{4.38}$$

Proof.
We show how the claim of the corollary follows from the second assertion in Theorem 4.2.6. (A proof based on the first assertion may be found in the proof of Kunisch and Volkwein 2002, Corollary 4.13.)
The constant c_P becomes one since the estimation of the norm of the projection in the $\mathcal{L}(V)$-norm is now possible, which is reflected in using $\|\cdot\|_{L^2(0,T;V)}$ instead of $\|\cdot\|_{L^2(0,T;H)}$ (see above).
The additive term depending on the non-modeled derivative may be estimated by a term depending on the non-modeled energy since we use an "extended" snapshot set (see above).
$\Delta t < T$, σ_n and the "coefficient" of λ_j may be estimated by the constant $C(T)$ (due to (4.37)).
Note that according to Proposition 2.3.9, the POD error expressions are bounded by their asymptotic version, which completes the proof. □

Dependencies in the Error Estimate We wish to stress the actual dependencies in the "improved" error estimated by introducing notation which simplifies the statements even further and then comment on the structure of the estimate and possible amendments.

Corollary 4.2.8 (Interpretation of Asymptotic Estimate)
Let the assumptions of Corollary 4.2.7 hold and define the Δt-independent errors of the (asymptotic) POD representation by:

$$I^\infty := \sum_{i=\ell+1}^{d(n)} |(\psi_i, y_0)_V|^2 \lambda_i^\infty \quad \text{and} \quad E^\infty := \sum_{i=\ell+1}^{d(n)} \lambda_i^\infty.$$

Then there exist constants $C(T)$ and $C_2(\|y_t\|_{L^2(0,T;H)}, \|y_{tt}\|_{L^2(0,T;V)})$, independent of ℓ and the grids Γ and Λ^ℓ, as well as $\overline{\Delta t} > 0$, depending on ℓ, such that for all $\Delta t \le \overline{\Delta t}$:

$$\sum_{k=0}^{m} \beta_k \|Y_k - y(\tau_k)\|_H^2 \le C\big(\underbrace{I^\infty + E^\infty}_{\text{POD error}} + \underbrace{C_2 \Delta\tau \Delta t + C_2(\Delta\tau)^2}_{\text{temporal error}}\big). \tag{4.39}$$

Remark 4.2.9 (Structure of Estimate)
As shown in (4.39), we may decompose the error of the reduced-order solution into the *spatial approximation error* of the Galerkin POD scheme as well as the *approximation error* of the temporal backward Euler scheme (just as in the proof of Theorem 4.2.6).
The type of the dependence of the temporal error on $\Delta\tau$ is induced by the implicit Euler method. We may obtain a dependence of higher order in $\Delta\tau$ by using a Crank Nicholson scheme for instance (and assuming appropriate regularity on y). (Refer for example to Kunisch and Volkwein 2001, Subsection 3.4 and Kunisch and Volkwein 2002, Remark 4.14.)

4.2.3 Variants of Error Estimates

So far, we have considered reduced-order models based on a POD Basis which was obtained for the choice $X = V$. In this subsection, we wish to very briefly investigate the cases $X = H$ on an abstract level, $X = L^2(\Omega)$ and $X = H^1(\Omega)$ on the level of parabolic IVP and finally $X = \mathbb{R}^q$, on an FE discretization level say. (Note that this choice also determines the *reference* solution of the reduced-order model; refer to Subsection 4.1.4.)

Application of POD with $X = H$ So far, we assumed to have applied the POD Method with $X = V$. There might however be good reason to apply the method with $X = H$ as we wish to optimally represent the snapshot ensemble in the H-Norm for example (refer to Remark 2.1.7 on the Optimality Norm).

It turns out that the analysis is less robust in this case: Inserting the H-error terms I_H^ℓ and E_H^ℓ for I^ℓ and E^ℓ in (4.39), respectively, the error estimate unfortunately depends on the norm of the "POD stiffness matrix" $\|S\|_2$. Yet this norm tends to infinity for an increasing number of time steps m. In particular, we find that, in contrast to the definitions in Corollary 4.2.8, the errors of the POD representation read:

Corollary 4.2.10 (POD Representation Error for $X := H$)
Let the assumptions of Corollary 4.2.7 hold, but choose $X = H$. Then, the Δt-*dependent* errors of the (asymptotic) POD representation are given by:

$$I_H^\ell(\Delta t) := \|S\|_2 \sum_{i=\ell+1}^{d(n)} (\psi_i, y_0)_V \quad \text{and} \quad E_H^\ell(\Delta t) := \|S\|_2 \sum_{i=\ell+1}^{d(n)} \lambda_i,$$

where $\|S\|_2$ denotes the *POD stiffness matrix*

$$S = ((S_{ij})) \in \mathbb{R}^{\ell \times \ell} \quad \text{with} \quad S_{ij} = a(\psi_j, \psi_i), \quad 1 \le i, j \le \ell.$$

Proof.
By using the fact $\|\varphi\|_V^2 \le \|S\|_2 \|\varphi\|_H^2$ and Corollary 2.3.4 with $X = H$, we infer that the POD error estimate for $X = H$ becomes (for every $\ell \in \{1, \dots, d\}$)

$$\sum_{j=0}^{n} \alpha_i \left\| y(t_j) - P^\ell y(t_j) \right\|_V^2 \le \|S\|_2 \sum_{i=\ell+1}^{d(n)} \lambda_i.$$

□

Application to Parabolic Problems For our particular example of parabolic IVP, we chose $V = H^1(\Omega)$ and $H = L^2(\Omega)$. Hence, for $X = V$ we have $\|S\|_2 = 1$, whereas for $X = H$ the spectral norm of S increases as ℓ increases.

However, since the H^1-norm includes both the L^2-norm as well as the gradient norm, the decay of the eigenvalues is not as fast as in the case $X = H$. Thus, the advantage of $\|S\|_2 = 1$ for $X = V$ is balanced by the disadvantage that for a given ℓ, the term E^ℓ is larger than the term E_H^ℓ for the choice $X = H$. – But if we choose ℓ in the way that is often chosen in practice, i.e., such that $\mathcal{E}(\ell)$ is lower than a given threshold (refer to Subsection 3.2.2), then the relative errors for $X = V$ are smaller than for $X = H$. This issue was discussed in detail in Hömberg and Volkwein 2003, p. 1016 for example.

Application to FE Solutions As mentioned in Subsection 4.1.4, our estimate covers the case of estimating the error of the FE-POD low order solution to the FE full solution. We simply choose $V = \mathbb{R}^q$. For an explicit derivation of estimates for this case, refer to Volkwein 2006, Section 2 and Hinze and Volkwein 2004, p. 6.

Let us stress again that the reference solution in this context is the FE solution – and not the actual (continuous) solution to the parabolic IVP or Evolution Problem.

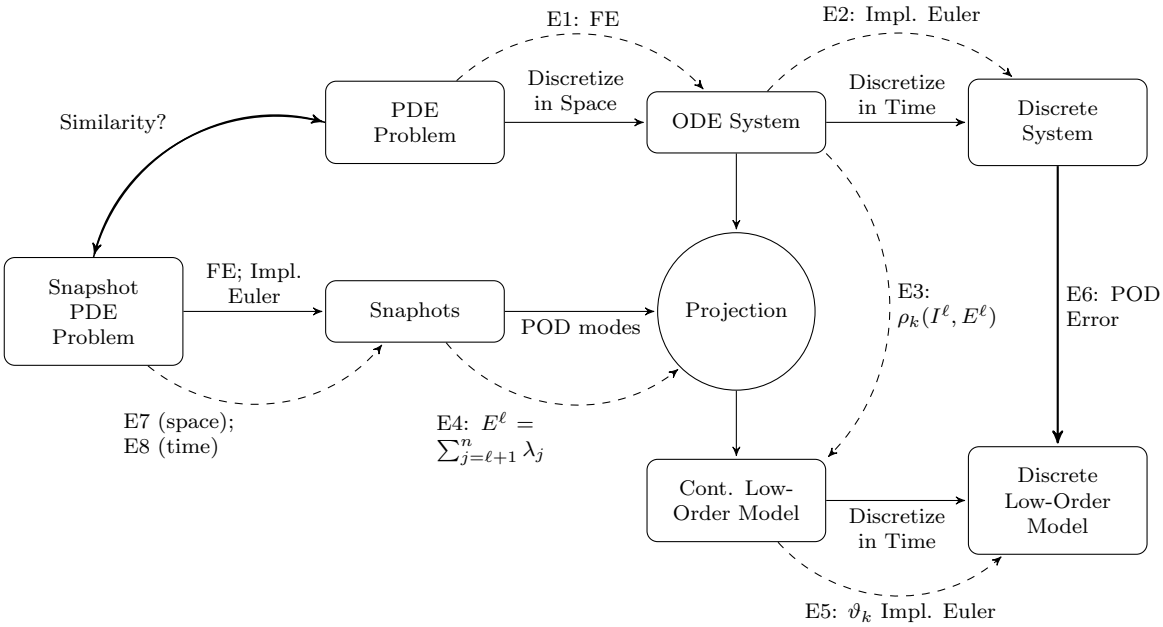

Figure 4.2: Sources of errors in a POD reduced-order model, including the possible perturbations in the snapshots in a practical context.

4.2.4 Perturbed Snapshots – Error Estimates in Practical Applications

In this subsection, we shortly wish to touch upon "perturbations" of the snapshots set, which may be due to two reasons:

1. The snapshots are taken from a system which is different from the system we want to establish a reduced-order model for.

2. The snapshots are obtained from a discrete approximation to the solution.

Therefore, the setting is somehow different from the error estimations of the previous subsections. The new situation is depicted in Figure 4.2.

Motivation In practice, the assumption to *know* snapshots of the solution is not realistic since we are interested in finding a solution at all by means of a reduced-order model. POD might however be very useful if we could obtain snapshots from an existing solution to one problem in order to setup a reduced-order model for another.

Estimation of Desire Given two system, suppose we obtain discretely approximated snapshots from system one and setup a reduced-order model for system two. The actual goal would be to present an error bound for the reduced-order solution system two in comparison to the *exact* solution of system two, depending on the correspondence of system one and two.
Unfortunately, this is well beyond the scope of this thesis. Thus, we choose to only comment on "reason 2", mentioned above, and assume that the two system coincide.

Errors due to Calculation of Snapshots We assume that the system we obtain the snapshots from and the system to setup a reduced-order model for coincide.
In the asymptotic analysis of Section 2.3, we have learned that arbitrary many *exactly known* snapshots lead to an "ideal" POD Basis ("in the limit").

In order to calculate snapshots however, we have to discretize the system in time and space (introducing the errors E7 and E8, compare Figure 4.2). In Subsection 3.2.4, we have found that the POD operator K_h converges at the rate of convergence of the FE approximation (refer to Volkwein 1999, Section 3), which takes care of E7. In Subsection 6.2.5, we also present a corresponding numerical study.
We have however also mentioned in Subsection 3.2.4 that we have not considered errors in the snapshots which are due to the time discretization (E8): It does make a difference whether we obtain snapshots from a numerical solution computed on a fine or a coarse time grid (also refer to the numerical example in Subsection 6.2.3).

4.3 Discussion of the POD as a Tool in Model Reduction

We shall conclude this section by a short discussion on the POD Method as a Model Reduction tool.

4.3.1 Optimality of the POD Basis in ROM

The fundamental idea of the POD Method is its optimality in representation, yet let us summarize *in which sense* the POD Method is "optimal".

Optimal Representation Due to the general construction of the POD Method in Section 2.1, there holds: For a given ensemble $\mathcal{V}_P$ and a rank ℓ, the POD Basis is an "in the quadratic mean" optimal basis, i.e., there is no other basis of lower rank that captures "on average" more information – "more" in the sense of the chosen norm $\|\cdot\|_X$ (refer also to Remark 2.1.7).
We shall stress that a POD Basis obtained from snapshots of an Evolution Problem is only an optimal representation of its rank for the *this very ensemble* – rather than of the Evolution System *itself*. (Refer to Remark 4.3.1 below.)

Asymptotic Analysis in Context of Evolution Problems Recall that in Section 2.3, we considered the convergence of the POD solution for an increasing number of (exactly known) snapshots. In context of Evolution Systems, this implies that we in the limit use the *whole trajectory* of the system as a snapshot set. In particular, we look for a POD Basis of an (naturally infinite) ensemble which consists of the whole trajectory $\{y(t) \mid t \in [0,T]\}$ of the solution $y : [0,T] \to X$ of the Evolution Problem. Therefore, we may think of this problem to be an *ideal POD Basis* since we have obtained it from an ensemble of *"all" snapshots* possible.
In fact, we desired a sequence of *finite* POD Problems to *converge* to such an *infinite* POD Problem by taking the number of snapshots to infinity. In this sense, we may think of the "ideal" POD Basis to be a "*limit*" basis.

Remark 4.3.1 (Optimality of POD Representation of Evolution Problems)
A POD *Limit* Basis is the best possible basis of its rank for the system whereas the "few snapshots" POD Basis is only optimal for the respective ensemble of snapshots. Hence, in case of the POD Limit Basis we can actually claim that this basis captures the essential information of the *problem* whereas a "few snapshots" POD Basis only captures the essential information of the respective *set of snapshots*.

Optimal Convergence Let us look at the previous issue more closely: According to Section 2.3, the representation of finitely many snapshots converges to the representation of the whole trajectory; at each step being optimal in the sense of the previous paragraph.
In this sense, for an increasing number of snapshots, the POD Method leads to an *optimally converging* sequence of "representations of snapshot sets" to the respective "representation of the Evolution Problem" (of which the snapshots have been taken).

4.3.2 Warnings

The Notion of Energy In Subsection 3.2.2, it was already mentioned that the POD Basis elements in the context of fluid dynamics (incompressible fluid mechanics with velocity measurements as snapshots) are related to the *modes of highest kinetic energy*. In particular, the eigenvalues of the POD Operator R (defined in (2.5)) denote the *energy contribution* of the respective POD mode. This interpretation however is not true in general. For example, in Chatterjee 2000, Subsection 6.4 the author provides an example of a physical system in which the *physical energy* shows no correlation with the "energy" of the POD Basis elements.

Mixed Variables in the Ensemble Due to Lemma 1.1.6, the scaling of the variables involved in a snapshot ensemble does matter. (SVD is not invariant under coordinate changes.) Hence, with inappropriate scaling of the respective variables, the POD Method may lead to an "optimal" representation of the snapshot set which is meaningless in terms of the full system of which the snapshots were taken (also refer to Chatterjee 2000, Subsection 6.1).
Therefore, we have to take care in particular if different variables are involved in a snapshot set – since these might well be of different scales. For example, it will turn out in Subsection 5.4.4 that it is a good idea to include snapshots of the state as well of as the adjoint state when performing *POD Suboptimal Control*. In this case, the POD Method so to say does not "know" whether a certain snapshot is taken from the state or the adjoint state.

4.3.3 Benefits of the POD Method

Huge Reduction in Computation Time The major goal of (POD) Model Reduction is of course to reduce the numerical costs of solving an Evolution Problem say – which is especially of importance in context of Optimal Control of such problems (refer to the further chapters).
Since a POD Basis is obtained from the system of consideration, it actually carries information about the system – in contrast to FE Ansatz functions say. Therefore, the size of the resulting system of ODEs may be decreased dramatically.
Actually, POD leads to an optimal construction of a ROM, based on snapshots taken from the respective system. It should not be concealed however, that – in terms of dimension reduction – there might be more effective approaches which choose snapshots more cleverly or which are not even based on information in snapshots (refer to Subsection 4.3.4 on the "drawbacks of POD").

Taking Advantage of the Linearity of the Method The POD Method is a linear procedure, but can be applied to non-linear systems as well (as the origin of the ensemble to be used is not taken into account at all). This on the one hand may be a benefit, but on the other hand this also may be a constraint as properties typical to non-linear systems cannot be represented at all (chaotic phenomena in turbulent flows, for instance).
However, let us mention a typical benefit: If the solution of a dynamical system is used as a snapshot ensemble, some of its properties are "inherited" by the POD Basis. For example, in an incompressible flow problem, the solution is supposed to be divergence free, i.e., the snapshots are divergence free and so are the POD Basis elements (due to the linearity of the POD method). This simplifies the reduced-order models considerably as the system will be projected on the (divergence free) POD Basis. In particular, the resulting system will not involve the pressure term and the divergence freeness condition anymore.

***Understanding* Structures in Solutions of Dynamical Systems** We shall learn in Chapter A that we may also make use of "form" of the basis functions themselves in order to actually gain an *understanding* of the dynamics of the respective system.

4.3.4 Comments on The Drawbacks of the POD Method

Snapshots Necessary Let us point out that in order to *obtain* snapshots at all, some sort of solution to the Evolution Problem has to be available. Even if we would assume to have access to snapshots at arbitrary time instances, it still is not straightforward to setup a suitable snapshot grid. Furthermore, in the analysis above, we have assumed that the snapshots are known exactly. In practice, the snapshots usually are perturbed in different ways (refer to Subsection 4.2.4). We did consider corresponding *asymptotic* estimates in the choice of the snapshot locations (Section 2.3) as well as in terms of the spatial approximation (refer to Subsection 3.2.4) – yet these estimates only teach us that "in the limit" the situation is fine. Hence, it is not clear in general *how* perturbations in the snapshots influence the error in the low-order solution.
In terms of "Suboptimal Control", the characteristics of the dynamics of a system might be changed significantly by changing the control and hence the snapshots become "worthless". (Refer to Subsection 5.4.5 for details on this problem of "non-modeled dynamics".)

Hard to Predict the Quality of a Low-order Solution To the author's knowledge, there is no reliable "procedure" to tell the quality of the respective reduced-order solution. I.e., there is no proper way to say *beforehand* how good a POD reduced-order solution will be. Typical questions which remain open would be for example:

- Given an Evolution Problem and an error bound, how to choose Γ and ℓ?
- Given an Evolution Problem, which choice of Γ minimizes the error of the reduced-order model?
- What is the influence of taking a certain amount of snapshots from a solution obtained on a *fine* time grid in comparison to taking snapshots from a solution computed on a *coarse* time grid?

In full generality, these questions are hard to tackle, although some effort is put into establishing an "optimal choice of snapshot grid"; keeping in mind that this necessarily increases the numerical costs.

Questionable Value of the POD "Optimality" POD is optimal only within an *a-posteriori data-analysis* scheme (refer to Subsection 4.3.1). There are *no guarantees* for the optimality in modeling. In particular, there could be models of even *lower* dimension which would capture the dynamics of a system much more precisely (for example if their input is better "suited" to the problem than the snapshots of choice are for POD).
In fact, given a snapshot set, the only parameter we may choose is ℓ, i.e., we may control how much information of the snapshots shall be contained in the POD Basis, the basis of the reduced-order model. This of course does not change the *value* (of the information contained in the snapshots) for setting up a reduced-order model. Therefore, we do not have explicit control over the "value" of information in the POD Basis towards modeling the actual solution.

Rank vs Information – Problem of Quickly Traveling Information The POD Method provides an optimal representation of parametrized data of some (desirably low) rank. We want to stress that the "rank of an approximation" is *not* to be confused with its "information content".
In context of Evolution Problems and the data being parameterized by space/time, the quality of the approximation decreases for example in the case that the "information" in the solution travels quickly with little "spread in space", i.e., *little correlation* between the snapshots.
On a matrix level, a simple example may illustrate this: Let two matrices contain the same information (i.e., "entries"). Yet in the first matrix all the information is concentrated in just one column, whereas in the second one the information is given on the diagonal, i.e., spread over all columns. In order capture all the information of the first matrix, just one basis vector would suffice, whereas for the second case, all columns are needed in order to capture all information.
Hence, we suppose that POD does *not* work explicitly well for problems whose solution characteristics *"travel quickly with little extension in space"* (see the "challenging example" in Chapter 6).

Chapter 5

(Sub) Optimal Control of Evolution Problems

In this chapter, we shall discuss a linear quadratic Optimal Control problem for the Evolution Problem 1.3.2 as well as its special case, the parabolic IVP 1.7. Then, a corresponding "suboptimal" problem shall be introduced (by means of the reduced-order model of Chapter 4). Finally, we give a short outlook on feedback control.

Procedure We give an intuitive idea of the problem of Optimal Control and phrase the (open loop control) problem of concern mathematically: A *convex*, linear-quadratic functional observes the final value as well as the full state of the system – and we seek to minimize its value by an "optimal control". For that purpose, we establish "optimality" on a continuous level.
In terms of *numerical treatment*, we propose two different ways of solution and introduce rather basic algorithms. (We actually wish to focus on suboptimal control strategies which shall not require sophisticated methods due to the small size of the system.)
We then apply the theory to a reduced-order model and discuss the treatment of the resulting "suboptimal control strategy".
We conclude with a brief outlook on so-called *feedback control* since this is a typical application of suboptimal control.
Corresponding numerical examples may be found in Section 6.3.

Literature Standard textbooks on the theory of Optimal Control problems of PDE are Lions 1971 as well as Troeltzsch 2006. A good introduction into numerical algorithms in context of Optimal Control may be found in Kelley 1999.
As far as Suboptimal Control is concerned, most of the theory is taken from Hinze and Volkwein 2005. (In particular, error estimates as well as experiments on the proper choice of snapshot set are given.) For a discussion of the "adaptive POD algorithm" refer to Hinze and Volkwein 2004 for example. For an extensive application of suboptimal flow control refer to the dissertation Bergmann 2004, for example.

5.1 Introduction to Optimal Control Problems

Intuitive Idea of Open-loop Control In Optimal Control of Evolution Problems (also known as Optimization of Systems governed by Evolution Problems), we essentially try to solve the following task: How to set parameters/control variables in an Evolution Problem such that a chosen objective is minimal?

Usually, this objective measures the agreement of some aspect of the state of the system with a desired one. The desired state is to be achieved by a suitable choice of the so-called *control variable* ("tracking control"). Alternatively, we could think the desired state to consist of "measurements" taken from the system. Then, we wish to determine "parameters" in the system, based on these measurements ("parameter estimation").

Open vs Closed-Loop Control In control theory, the procedures described above are referred to as *Open Loop Control*. From a practical point of view, this concept is explicitly useful for *long-term* type of problems – designing the layout of the airflow in an airplane, for instance. This operation is to be carried out once and as exact as possible.
Once the plane is built, one might also wish to steer the airflow to an actually desired state, i.e., a comfortable condition for the passengers. This process should be immediate and in response to observations of the state, i.e., the temperature distribution during the flight at a given time instance. A typical question to answer could be: Where to best increase the temperature according to the model?
This presents the motivation for the issue of so-called *Feedback-* or *Closed Loop Control*: We do not calculate a single control such that (together with the resulting state) an objective is minimized, but try to discover the *dependence* of the optimal control on measurements of the state.

The Problem of Feasibility Although being highly relevant in a vast amount of applications, solving Optimal Control problems is in general not "a piece of cake" – even for modern computers. For a more complex model, even a "forward simulation" may already present a challenge, yet the effort demanded by Optimal Control problems is even higher.

5.2 Linear-Quadratic Open Loop Control of Evolution Problems

In this section, we wish to mathematically state the Optimal Control problem of concern, investigate the existence of an optimal solution and derive respective optimality conditions. (For more details on the theory of the respective Evolution Problem refer to Section 1.3.)

General Open-Loop Control Problem As depicted in Figure 5.1, a control problem for an Evolution Problem generally consists of the *objective* (or *cost functional*), the *state* and the *control*. The value of the objective depends on the state and the control (state and control "observation"). The aim of solving the optimal control problem is to find a control such that the value of the objective is minimized (or maximized). The *dependence* of the state on the control is determined by the "Evolution Problem constraint". Additionally, constraints on the actually possible values of the control as well as of the state may be imposed.
Optimal control problems may then be classified by the *type of objective*, the type of Evolution Problem constraint as well as the type of control and the type of observations. In terms of control of parabolic IVP, there are three basic types of controls: distributed control, boundary control and initial value control.

5.2.1 Mathematical Problem Statement

Having set up the general context of an optimal control problem, let us define the particular ingredients of the problem of concern. Note that we will derive our actual problem of concern from the more general case introduced in Lions 1971, Section III.2. (Since we shall quote proofs from this reference, that procedure shall help to match the situations. Furthermore, the roles of state and space observation are more obvious in these statements.)

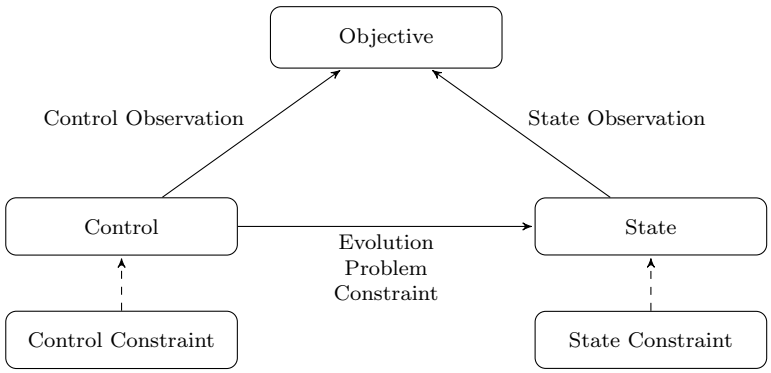

Figure 5.1: General layout of an open-loop control problem.

Note that we shall *not* consider "initial value control". The other "types of control" ("distributed" or "boundary") are not determined at this stage of "Evolutions Problems" and hence are both covered.

Convex Control Space Let $\mathcal{U}$ be a Hilbert space which we identify with its dual $\mathcal{U}'$ and let $\mathcal{U}_{ad} \subset \mathcal{U}$ be a closed and *convex*, nonempty subset. We call the restrictions defining $\mathcal{U}_{ad}$ *"control constraints"*.

Evolution Problem Constraint Let us first define the continuous linear *"control operator"* $\mathcal{B} : \mathcal{U} \to L^2(0,T;V')$. We also introduce its (linear and bounded) dual operator $\mathcal{B}^* : L^2(0,T;V) \to \mathcal{U}' \sim \mathcal{U}$ satisfying

$$(\mathcal{B}u, \phi)_{\mathcal{L}^2(0,T;V'),L^2(0,T;V)} = (\mathcal{B}^*\phi, u)_{\mathcal{U}} \quad \text{for all } (u,\phi) \in \mathcal{U} \times L^2(0,T;V).$$

For $y_0 \in H$, $u \in \mathcal{U}_{ad}$ and $F := \mathcal{B}u$ the linear Evolution Problem 1.3.2 reads (for all $\phi \in V$):

$$\frac{d}{dt}(y(t),\phi)_H + a(y(t),\phi) = ((\mathcal{B}u)(t),\phi)_{V',V}, \quad t \in (0,T] \text{ a.e.,} \tag{5.1a}$$

$$(y(0),\phi)_H = (y_0,\phi)_H. \tag{5.1b}$$

State Space In the Evolution Problem 1.3.2, we only have set $F = \mathcal{B}u \in L^2(0,T;H)$. Hence, we infer from Proposition 1.3.3 that (for every $u \in \mathcal{U}$ and $y_0 \in H$,) there exists a unique weak solution $y \in W(0,T)$ of (5.1). Thus, as a state space, we may choose $W(0,T)$. Of course, one might wish to impose further constraints on the state which we shall refrain from. (For a discussion of state-constraint problem refer to Lions 1971, for example.)

State and Control Observation In order to setup the objective functional, it remains to introduce a state as well as a control observation, i.e., the dependence of the objective on the state as well as the control variable. For that reason, let us introduce Hilbert spaces $\mathcal{H}$, $\mathcal{H}_1$ and $\mathcal{H}_2$ which we call *"observation spaces"*.
For $u \in \mathcal{U}$, a "general" *state observation* is given by $C \in \mathcal{L}(W(0,T);\mathcal{H})$, $z(u) = Cy(u)$. Unfortunately, the treatment of this case would involve the dual space of $W(0,T)$ which would require considerations which are somewhat complicated. Hence, it is better to consider two cases: either observing the *whole trajectory* of the state or observing the *final state* only. (We shall then use a linear combination of these two cases.) In particular, we choose

$$C_1 \in \mathcal{L}(L^2(0,T;V);\mathcal{H}_1) \quad \text{and} \quad C_2 \in \mathcal{L}(L^2(0,T;V),\mathcal{H}_2) \text{ with } C_2 y = Dy(T), D \in \mathcal{L}(H,\mathcal{H}_2).$$

A *control observation* generally is given by (for a constant $\nu > 0$):

$$N \in \mathcal{L}(\mathcal{U},\mathcal{U}), \quad (Nu,u)_{\mathcal{U}} \geq \nu \|u\|_{\mathcal{U}}^2, \quad u \in \mathcal{U}.$$

General Linear Quadratic Objective Functional We choose the "general" state observation C to be a linear combination of C_1 and C_2 and choose $z_1 \in \mathcal{H}_1$, $z_2 \in \mathcal{H}_2$. Then, a linear quadratic *"cost functional"* is given by

$$J(y,u) = \frac{\alpha_1}{2} \|C_1 y - z_1\|^2_{\mathcal{H}_1} + \frac{\alpha_2}{2} \|C_2 y - z_2\|^2_{\mathcal{H}_2} + \frac{\sigma}{2} (Nu, u)_{\mathcal{U}} . \tag{5.2}$$

Note that by means of the triangle inequality and Young's inequality, we may show that this functional is *convex*.

Actual Cost Functional We choose $\mathcal{H}_1 := L^2(0,T;H)$ and $\mathcal{H}_2 := H$. Let $z_1 \in L^2(0,T;H)$ be a *desired trajectory* of the state y and let $z_2 \in H$ be a *desired final state*. For the sake of simplicity, we simply choose C_1 to be the injection map of $L^2(0,T;V)$ into $L^2(0,T;H)$. Furthermore, let D and N be the identities on H and $\mathcal{U}$, respectively. (For a more complex choice, refer to Hinze and Volkwein 2005, Remark 2.2, for example.)
From (5.2), we may then deduce the form of our linear quadratic cost functional $J : W(0,T) \times \mathcal{U} \to \mathbb{R}$:

$$\begin{aligned} J(y,u) &= \frac{\alpha_1}{2} \|y - z_1\|^2_{L^2(0,T;H)} + \frac{\alpha_2}{2} \|y(T) - z_2\|^2_H + \frac{\sigma}{2} (u,u)_{\mathcal{U}} \\ &= \frac{\alpha_1}{2} \int_0^T \|y(t) - z_1(t)\|^2_H \, dt + \frac{\alpha_2}{2} \|y(T) - z_2\|^2_H + \frac{\sigma}{2} \|u\|^2_{\mathcal{U}} . \end{aligned} \tag{5.3}$$

The first additive term measures the agreement of the trajectory $y(t)$ with z_1 whereas the second one measures the agreement of $y(T)$ and z_2. The last additive term accounts for the control cost involved in the problem. The parameters α_1, α_2 and σ decide on the importance of the respective contributions towards the total cost. In particular, σ is also of some importance as it denotes a sort of *stabilization parameter* as well (refer for example to Volkwein 2006).

The Actual Control Problem Combining all this work, we may concisely state the optimization problem of concern:

$$\min J(y,u) \qquad \text{s. t.} \qquad (y,u) \in W(0,T) \times \mathcal{U}_{ad} \quad \text{solves} \quad (5.1). \tag{OC}$$

By means of the solution operator S (refer to Definition 1.3.4), we may state (OC) in its *reduced form*:

$$\min_{u \in \mathcal{U}_{\text{ad}}} \hat{J}(u) := J(Su, u) \tag{ROC}$$

Existence of a Solution Finally, let us quote a result that the problem (OC) actually admits a solution.

Proposition 5.2.1 (Existence of Optimal Solution)
There exists a unique optimal solution $\bar{x} = (\bar{y}, \bar{u})$ to (OC).

Proof.
Refer to Lions 1971, (2.10). □

5.2.2 Theory for Optimal Control

Since we wish to state optimality conditions to problem ROC, we quote a corresponding general lemma. As this lemma involves the derivatives of the cost function (which in our case is defined on a Hilbert space), we introduce a respective understanding of a derivative in this context.

Differentiation on Banach Spaces Since we aim to minimize a functional on a Banach space, we shall introduce the respective notation.

Definition 5.2.2 (First Variation, Gateaux Derivative)
Let U be a real Banach space and $F : U \to \mathbb{R}$ a functional on U. If for $u, h \in U$ the following limit exists, we define the *first variation* of F to be

$$\delta F(u,h) := \lim_{t \searrow 0} \frac{1}{t}(F(u+th) - F(u)).$$

Asumme $u \in U$. If for all $h \in U$, the first variation $\delta F(u,h)$ and an operator $A \in U^*$ exist such that

$$\delta F(u,h) = Ah,$$

we say that A is the *Gateaux-derivative* of F in u.

Basic Characterization of Optimal Solution We may now quote the result that we shall derive the optimality conditions from. (Since we deal with a convex set of admissible controls $\mathcal{U}_{\text{ad}}$ and a convex objective functional the following, in general only necessary condition, also is *sufficient*.)

Lemma 5.2.3 (Variational Inequality)
Let U be a real Banach Space, $C \subset U$ a *convex* set and $f : C \to \mathbb{R}$ a real-valued *convex* functional which is Gateaux-differentiable on C. $\bar{u} \in C$ is a solution of

$$\min_{u \in C} f(u)$$

if and only if there holds

$$f'(\bar{u})(u - \bar{u}) \geq 0 \quad \text{for all } u \in C.$$

Proof.
Refer to Troeltzsch 2006, Lemmas 2.20 and 2.21. □

5.2.3 Optimality Conditions in an Abstract Setting

As mentioned in the introduction, there are different numerical approaches to solving Optimal Control problems (refer to Section 5.3). In view of the approach "optimize-then-discretize", let us establish optimality conditions for problem (5.2) on a continuous level.

Procedure We apply Lemma 5.2.3 to the reduced form of the optimization problem (ROC). It turns out that the derivative depends on the adjoint S^* of the solution operator S.
In order to establish a formulation for S^*, we view the full control problem (OC) as a *constrained* optimization problem in *two* variables (i.e., the state and the control). We apply the Lagrangian approach and find that the resulting Lagrange parameter is the solution to an Evolution Problem whose solution operator is given by S^*.
This idea of procedure is illustrated in Figure 5.2 and we shall now walk along its two major paths.

Situation Considering (OC), we are given a smooth *convex* functional $J(u,y)$ in a Hilbert space depending on two variables u and y which are not independent of each other due to the IVP constraint. Furthermore, there are constraints on the possible values of u.

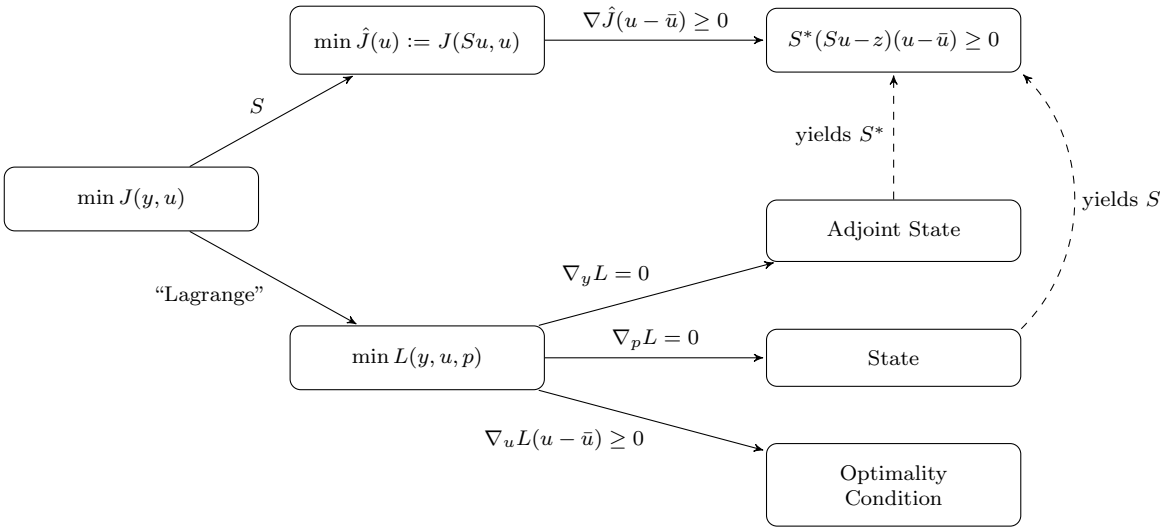

Figure 5.2: Idea of establishing an optimality system. (Note that the formula in the upper right node is simplified for the sake of suitable presentation.)

Reduction to One Variable By means of the (abstract) solution operator S, we may express y in terms of u and are hence able to introduce $\hat{J} := J(u, S(u))$ (refer to (ROC)). Due to the Variational Lemma 5.2.3, we only need to look for the points where the derivative is non-negative in all admissible directions (since the functional is convex).
Thus, the essential task is to establish the derivative of $\hat{J}$. This is not straightforward since it involves the solution operator S of an Evolution Problem.

Establishing the Derivative Let us start with the basic case of a functional $f(u) := \|u\|_H^2$, defined on a Hilbert space H. For the derivative of f, we obtain:

$$\langle f', h \rangle_{H',H} = (2u, h)_H \quad \text{or} \quad f'(u) = 2u,$$

where the latter alternative is called the *"gradient"* of f and obtained by identifying H with its dual H^* and the well known Proposition of Riesz.
Returning to our actual problem, note that $\hat{J}$ is composed of terms of the structure

$$E(u) = \|Su - z\|_X^2 \quad \text{with} \quad E'(u) = 2S^*(Su - z), \tag{5.4}$$

where the gradient is to be understood in the sense above and may be obtained by a straightforward calculation (refer to Troeltzsch 2006, (2.38)). Thus, it remains to establish a suitable statement of S^*.

Alternative Approach via Lagrange Technique In the approach above, we have incorporated the Evolution Problem constraint by expressing y in terms u (by means of S) and hence, the constraint has to be fulfilled for the S-linked pair $(u, S(U))$. In order to establish a formulation for the operator S^*, we now think of no variable to be "dependent" on the other – the pair (u, y) just has to satisfy the Evolution Problem.
We may then use the well-known Lagrange-technique for constrained optimization problems in order to solve the problem. (For mathematical details on the technique refer to Troeltzsch 2006, Subsection 6.1.1, for example.)
In order to setup the *Lagrange Functional* $L(y, u, p)$, we expand our objective J by the Evolution Problem constraint, weighted by a *Lagrange Multiplier* p.

Note that there are no constraints on y or p but there are on u. Thus, if $(\bar{y}, \bar{u}, \bar{p})$ is an optimal point $D_y L$ and $D_p L$ have to vanish in this point and $D_u L$ has to fulfill a variational inequality of the type introduced in Lemma 5.2.3.

Adjoint Problem Analyzing this in more detail, we find that $D_p L$ simply gives the Evolution Problem constraint, namely (a weak formulation of) the state equation and $D_u L$ yields the *optimality condition* from above.
The information we are after is "decoded" in $D_y L$. A careful calculation shows that this constraint might be interpreted as a weak formulation of the so-called *adjoint problem* (This is shown nicely in Troeltzsch 2006, Section 3.1 for example.) The solution of this problem actually yields the Lagrange multiplier p (refer to Troeltzsch 2006, Section 2.13). Furthermore, the respective solution operator is given by S^*.

Calculating the Derivative Since S^* is the solution operator of the "adjoint problem", we may state the derivative of $\hat{J}$ by means of this problem. Note that there are three additive terms of type (5.4) in $\hat{J}$, involving the operator S^*. Treating the control term is straightforward and the other two terms are treated by setting up adjoint problems with suitable data. We may combine these linear problems linearly which leads to a linear combination of the solutions. Hence, we expect the derivative to consist of two summands (see below).

Optimality Condition We have now established all ingredients and may summarize our findings in the following proposition. (A detailed version of the proof might be found in Lions 1971, Theorem 2.1, Theorem 2.2, for example. There the cases $C = C_1$ and $C = C_2$ are treated individually. A combination of those two cases is then straightforward.)

Proposition 5.2.4 (First Order Optimality Condition)
The pair $\bar{x} = (\bar{y}, \bar{u})$ is the (unique) solution of problem (OC) if and only if $\bar{x}$ fulfills the state equation (5.1) and with the unique Lagrange-multiplier $\bar{p} \in W(0,T)$ satisfies (for all $\phi \in V$) the following *adjoint equation* in $[0,T]$

$$-\frac{d}{dt}\left(\bar{p}(t), \phi\right)_H + a(\bar{p}(t), \phi) = \alpha_1 \left(z_1(t) - \bar{y}(t), \phi\right)_H, \quad \text{for all } t \in [0,T] \text{ a.e.,} \tag{5.5a}$$

$$\left(\bar{p}(T), \phi\right)_H = \alpha_2 \left(z_2 - \bar{y}(T), \phi\right)_H \tag{5.5b}$$

as well as the optimality condition:

$$\left(G(\bar{u}), u - \bar{u}\right)_{\mathcal{U}} \geq 0 \quad \text{for all } u \in \mathcal{U}_{ad}, \tag{5.6}$$

where the operator $G : \mathcal{U} \to \mathcal{U}$ is defined by

$$G(u) = \hat{J}'(u) = \sigma u - \mathcal{B}^* p. \tag{5.7}$$

Proof.
Introducing the adjoint state (5.5), we may transform the statement of the gradient of $\hat{J}$ at $\bar{u}$ into

$$\hat{J}'(\bar{u}) = \sigma \bar{u} - \mathcal{B}^* \bar{p}, \tag{5.8}$$

where y = y(u) solves the state equations (5.1) with the control $u \in \mathcal{U}$ and $p = p(y(u))$ satisfies the adjoint equations (5.5) for the state y (refer to Lions 1971, Subsection III.2.3). (Note that according to Lions 1971, p. 113 the adjoint state p is uniquely determined.)
The uniqueness of the solution $\bar{x}$ follows from Proposition 5.2.1. □

Unconstrained Case In the case that there are no constraints on the control, i.e., $\mathcal{U}_{\text{ad}} = \mathcal{U}$, we find that the optimality condition of Proposition 5.2.4 reduces to a *coupled system* of Evolution Problems – of which one is forward and one is backward in time.

Corollary 5.2.5 (Unconstrained Optimal Conditions)
Set $\mathcal{U}_{\text{ad}} := \mathcal{U}$. Let the pair $\bar{z} = (\bar{y}, \bar{p}) \in W(0,T) \times W(0,T)$ fulfill the following "system" of Evolution Problems (for all $\phi \in V$)

$$\begin{aligned} \frac{d}{dt}\left(\bar{y}(t),\phi\right)_H + a(\bar{y}(t),\phi) &= \left((-\sigma^{-1}\mathcal{B}\mathcal{B}^* p))(t),\phi\right)_{V',V} \quad \text{for all } t \in [0,T] \text{ a.e.,} \\ \left(\bar{y}(0),\phi\right)_H &= \left(\bar{y}_0,\phi\right)_H \end{aligned}$$

and

$$\begin{aligned} -\frac{d}{dt}\left(\bar{p}(t),\phi\right)_H + a(\bar{p}(t),\phi) &= \alpha_1\left(z_1(t) - \bar{y}(t),\phi\right)_H \quad \text{for all } t \in [0,T] \text{ a.e.,} \\ \left(\bar{p}(T),\phi\right)_H &= \alpha_2\left(z_2 - \bar{y}(T),\phi\right)_H . \end{aligned}$$

Then, there holds: The pair $\bar{x} = (\bar{y}, \bar{u})$ is the (unique) solution of problem (OC) if and only if

$$\bar{u} := -\frac{1}{\sigma}\mathcal{B}^*\bar{p}.$$

Proof.
The optimality condition (5.6) reduces to the equation

$$G(u) = \sigma u + \mathcal{B}^* p = 0 \quad \text{and hence,} \quad u = -\frac{1}{\sigma}\mathcal{B}^* p$$

may be substituted in the state equation. Together with the adjoint equation, we obtain the assertion. □

5.2.4 Application to Parabolic Problems of Second Order

Let us apply the theory developed to a control problem whose state equation is given by the parabolic IVP whose strong statement is given by Problem 1.3.5. Looking at the corresponding weak formulation (Problem 1.3.6), we may easily see how to deduce the optimization results for this case from the findings above. In this more concrete setting, we may now distinguish between the cases of "boundary control" and "distributed control". We shall however focus on the latter case. (Note that throughout, we shall use the notation introduced in Section 1.3; such as $V := H_0^1(\Omega)$.)

Distributed Control Problem In Problem 1.3.5, we simply choose the right hand side F to be our control u. (i.e., in terms of the theory above, we have chosen $\mathcal{B}$ to be the identity.) In strong form, the corresponding Optimal Control problem then reads:

Problem 5.2.6 (Distributed IVP Control)
For $V := H_0^1(\Omega)$, $H := L^2(\Omega)$, $u_a, u_b \in \mathcal{U} := L^2(Q)$ and

$$\mathcal{U}_{\text{ad}} := \{u \in \mathcal{U} \mid u_a(x,t) \le u(x,t) \le u_b(x,t) \text{ a.e. in } Q\} \subset \mathcal{U}$$

we wish to

$$\min_{u \in \mathcal{U}_{\text{ad}}} J(y,u) = \frac{\alpha_1}{2}\int_0^T \|y(t) - z_1(t)\|_H^2 \, dt + \frac{\alpha_2}{2}\|y(T) - z_2\|_H^2 + \frac{\sigma}{2}\|u\|_{\mathcal{U}}^2$$

such that:

$$\frac{\partial y}{\partial t} - \sum_{i,j=1}^{n} \frac{\partial}{\partial x_i}\Big(a_{ij}\frac{\partial y}{\partial x_j}\Big) + cy = u \qquad \text{in } Q_T, \tag{5.9a}$$

$$y(t,x) = 0 \qquad \text{on } (0,T)\times\partial\Omega, \tag{5.9b}$$

$$y(0,x) = y_0(x) \qquad \text{in } \{0\}\times\Omega. \tag{5.9c}$$

Optimality Conditions The parabolic IVP is just a special case of the linear Evolution Problem considered above (refer to its weak formulation in Problem 1.3.6). Thus, we may easily state optimality conditions by means of the results of the previous subsection.

Corollary 5.2.7 (Optimality Conditions for IVP Control)
The pair $\bar{x} = (\bar{y}, \bar{u})$ is the (unique) solution of Problem 5.2.6 if and only if $\bar{x}$ fulfills the state equation (5.9) and with the unique Lagrange-multiplier $\bar{p} \in W(0,T)$ satisfies the *adjoint equation* in $[0,T]$

$$-\frac{d}{dt}\bar{p}(t) - \sum_{i,j=1}^{n} \frac{\partial}{\partial x_i}\left(a_{ij}\frac{\partial \bar{p}}{\partial x_j}\right) + c\bar{p} = \alpha_1(z_1(t) - \bar{y}(t)) \qquad \text{in } Q_T,$$

$$\bar{p}(t,x) = 0 \qquad \text{on } (0,T)\times\partial\Omega,$$

$$\bar{p}(T) = \alpha_2(z_2 - \bar{y}(T)) \qquad \text{in } \{0\}\times\Omega$$

as well as the optimality condition

$$(G(\bar{u}), u - \bar{u})_{\mathcal{U}} \geq 0 \quad \text{for all } u \in \mathcal{U}_{ad},$$

where the operator $G : \mathcal{U} \to \mathcal{U}$ is defined by

$$G(u) = \sigma u - p. \tag{5.10}$$

Proof.
Effectively, we only need to check that the given adjoint state is the respective concrete case of the one in Proposition 5.2.4. But this is straightforward (for details, refer to Lions 1971, Section 3.1). The result is also provided in Troeltzsch 2006, Subsection 3.6.4. □

5.3 Numerical Treatment

In this section, we wish to explore the possibilities of tackling the optimal control problem (OC) (as well as its special case, Problem 5.2.6) numerically. In particular, we investigate two approaches (optimization on a "discrete level" vs optimization on a "continuous level"). For both of them, we consider "control constrained" as well as "unconstrained" cases.

Discretization of Optimal Control Problems The discretization of Optimal Control problems is not straightforward since there are quite a few "ingredients" to be discretized: The control space (with its constraints), the state space, the adjoint state space as well as the objective functional. Of course, all these objects are linked and hence the discretizations in some sense have to be "compatible" with each other. Therefore, there are special concepts on how to establish discretization schemes for

Optimal Control problems (refer for example to Hinze 2004). Apart from that, there has been quite an array of discussions on which procedure to carry out first – the optimization or the discretization.

Possible Ways of Solution Choosing to "discretize" first, we discretize the Evolution Problem constraint (i.e., discretize S so to say) as well as the objective functional J in order to obtain a linear "quadratic programming problem". This may be solved by standard routines. (If there are no constraints on the control, this problem actually reduces to a system of linear equations.)
Choosing to optimize first, we consider a "gradient projection method" in order to solve the (discretized) optimality conditions of Proposition 5.2.4. (Each cycle in the algorithm involves the solution of two Evolution Problems.) In the unconstrained case, our optimality condition becomes a coupled system of Evolution Problems (Corollary 5.2.5). After discretizing this system, the solution is characterized by a linear system of equations.

5.3.1 "Discretize" Then "Optimize" – Quadratic Programming

Let us shortly explain how to tackle an optimization problem by choosing a basis for the control and solving the optimal control problem in the respective coefficients. Clearly, this leads to a problem of finite dimension and may be tackled without the theory of Subsection 5.2.3.
For the sake of simplicity, we shall assume that we only need to discretize the control space and that we can then solve the state equation and calculate all integrals appearing exactly. (More details on the matter may be found in Troeltzsch 2006, Subsection 3.7.2, for example.)

Discretization of Control Suppose we are given a "discretization" of the control by

$$u(x,t) = \sum_{i=1}^{M} u_i e_i(x,t).$$

This discretization may for example be obtained from an FE approximation in space and an Euler method in time. The resulting coefficient "matrix" for u (each column consists of an FE vector) might then be reshaped to a single column vector.

Problem Setup We insert the ansatz for u into the objective functional and "optimize" on the coefficients of the ansatz only. We may now show that the optimality problem (for constraints as in Problem 5.2.6,) leads to a system of the type

$$\min\big(a^T u + \frac{1}{2} u^T C u + \frac{\lambda}{2} u^T D u\big), \quad u_a \leq u \leq u_b \tag{5.11}$$

(refer Troeltzsch 2006, (3.59), for example). This is a standard "quadratic programming" problem and may be solved by standard optimization routines (such as "quadprog" in Matlab).

Unconstrained case In case that there are no constraints on the control, we may deduce optimality conditions by basic "calculus of matrices". Setting $A = C + \sigma D$, the optimality conditions read

$$\nabla f = A\bar{u} = a \quad \text{and} \quad \nabla^2 f = A. \tag{5.12}$$

Therefore, the optimal solution is in this case given by the solution to a system of linear equations.

Feasibility As mentioned in the introduction, solving the resulting system is unfortunately very expensive (in the unconstrained as well as the constrained case): In general, A is dense and huge (it has $(N_{\mathrm{FE}})^2$ entries for each time step). Furthermore, A is expensive to assemble since for each entry an IVP has to be solved.
Obviously, this gives rise to *suboptimal* control strategies.

5.3.2 "Optimize" Then "Discretize" – Gradient Projection Method

Choosing to "optimize" first, we shall obtain "optimality conditions" on a continuous level (Proposition 5.2.4). Then, we discretize these condition and illustrate a possible way of solution by a gradient projection method. (Since we proceed different in case that there a no constraints imposed on the control, we take care of this matter in Subsection 5.3.3.)

Gradient Projection vs Newton's Method Newton's method basically tries to find roots of a functional, converges only locally in general, but ("close" to a root) of second order. There are *"Newton-type"* methods to tackle "variational equation" such as in the optimality condition (5.6).
Since the optimality condition of Proposition 5.2.4 involves the gradient of the objective functional, a Newton-type method needs the Hessian of the objective functional. But for the linear quadratic functional of concern, the Hessian is easy to access. Furthermore, the functional is convex and hence the algorithm converges globally. (For an application of Newton's method to optimal control of PDE, refer to the diploma thesis Schütz 2007, for example.)
Consequently, in practice, we surely would prefer to apply a Newton-type rather then a gradient projection method. Yet as outlined above, we do not want to focus on particularly efficient algorithms to tackle Optimal Control problems but wish to illustrate how optimal control problems may be tackled. (Anyway, we plan to reduce the problem size via a Suboptimal Control approach. Hence, slow convergence properties of the algorithm become less important since each iteration step should become rather simple to carry out.)

Idea of the Gradient Projection Algorithm The gradient projection algorithm is an extension of the *steepest descent algorithm* to "constrained problems" (refer to Kelley 1999, Section 3.1) The fundamental idea is that the "anti-gradient" of an objective functional f (i.e., $-\nabla f$) points into the direction of "steepest descent".
For a control x, a step length λ and a projection $\mathcal{P}$ that ensures that the set of admissible controls is not left, one iteration of the algorithm reads:

$$x_{n+1} = \mathcal{P}\big(x_n - \lambda \nabla f(x_n)\big).$$

Armijo Rule It only remains to establish a procedure for determining the step length λ. For $\beta \in (0,1)$, let $m \geq 0$ be the smallest integer such that for $\lambda = \beta^m$, there is "sufficient decrease" in f. In this context, this shall mean that

$$f(x_c - \lambda \nabla f(x_c)) - f(x_c) < -\alpha\lambda \left\|\nabla f(x_c)\right\|^2.$$

Further discussion of this procedure as well as possible improvements are given in Kelley 1999, Section 3.2.

Further Ingredients Note a stopping criterion like $\hat{J}'(u_{n+1}) < tol$ is not suitable since we use constraints on the control and hence the gradient does not have to vanish in the optimal point. Therefore, we choose to break the iteration if the "change in the control" is sufficiently small. In particular, we consider the "effect" of the step length. I.e., we employ a so-called *measure of stationarity*, which is a simplified version of Kelley 1999, (5.18).
Furthermore, we choose $\mathcal{U}_{\text{ad}}$ to be defined by vectors u_a and u_b (analogously to Problem 5.2.6) and denote the corresponding projection by $P_{[u_a,u_b]} : \mathcal{U} \to \mathcal{U}_{\text{ad}}$.
Finally, the gradient of the functional J (which is to be used as a direction of descent) was computed in Proposition 5.2.4 and reads (compare (5.6))

$$\hat{J}'(u_n) = \sigma u_n + \mathcal{B}^* p_n.$$

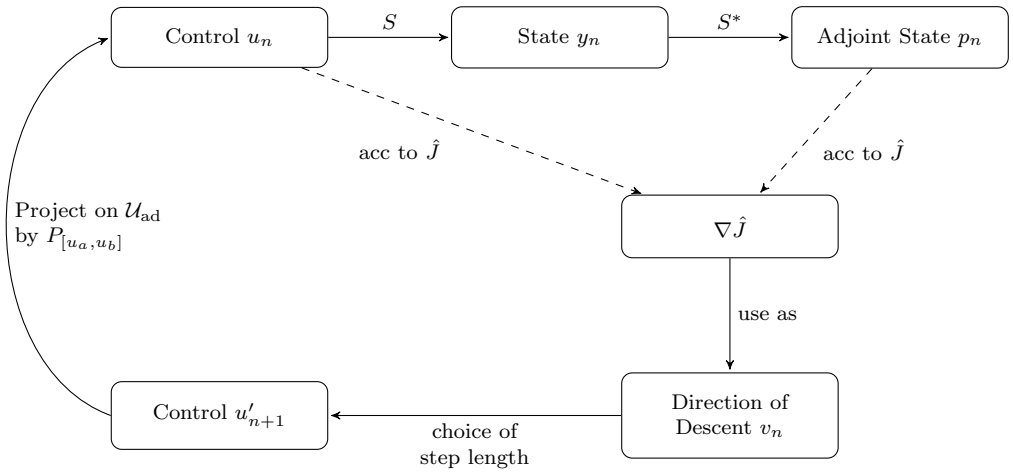

Figure 5.3: One cycle of the gradient projection algorithm for Optimal Control of an Evolution Problem.

Actual Realization We may now gather all our findings and formally state the gradient projection algorithm. Note that one cycle of this gradient projection algorithm is depicted in Figure 5.3. (More details on the algorithm as well as an application may be found in Troeltzsch 2006, Subsection 3.7.1. For a thorough discussion as well as a convergence analysis, refer to Kelley 1999, Subsection 5.4.2 or Sachs and Gruver 1980.)

Algorithm 5.3.1 (Gradient Projection)
Choose a "tolerance" tol as well as an "initial control" u_0. Then proceed as follows:

S1: State Determine the state y_n belonging to the control u_n by solving the state equation (5.1).

S2: Adjoint State Determine the adjoint state p_n belonging to the state y_n by solving the adjoint state equation (5.5).

S3: Direction of Descent Choose the Anti-Gradient as a *direction of descent*

$$v_n := -\hat{J}'(u_n) = -(\sigma u_n + \mathcal{B}^* p_n).$$

S4: Step Length By the Armijo rule, determine the *optimal step length* s_n from

$$\hat{J}\big(P_{[u_a,u_b]}(u_n + s_n v_n)\big) = \min_{s>0} \hat{J}\big(P_{[u_a,u_b]}(u_n + s\, v_n)\big).$$

S5: Update Control Set $u_{n+1} := P_{[u_a,u_b]}(u_n + s_n v_n)$.

S6: Termination If $\big\|u_n - P_{[u_a,u_b]}(u_n + 1v_n)\big\| > \text{tol}$, set $n := n+1$ and go to S1.

5.3.3 "Optimize" Then "Discretize" – No Control Constraints

On a continuous level, we have found that, with no control constraints, the optimality system is given by a system of two coupled Evolution Problems (see Corollary 5.2.5).
For the case of parabolic IVP, let us in this subsection derive a discrete approximation to that system – benefiting from the results of Section 1.4. (This approach is actually used in the numerical examples in Section 6.3.)

Space-Time-Discretization of parabolic IVP Suppose we discretize the parabolic IVPs (state as well as adjoint state equation) by means of an FE approximation with q degrees of freedom in space and apply the implicit Euler method with M time steps in time.
In Subsection 1.4.2, we have shown that we may obtain the FE coefficients C^m at time step m by solving a linear system of the form (refer to (1.18))

$$\begin{pmatrix} I & & & & \\ -D & K_0 & & & \\ & -D & K_1 & & \\ & & \ddots & \ddots & \\ & & & -D & K_{M-1} \end{pmatrix} \begin{pmatrix} C^0 \\ C^1 \\ C^2 \\ \vdots \\ C^M \end{pmatrix} = \begin{pmatrix} \tilde{g} \\ \tau_1 F^1 \\ \tau_2 F^2 \\ \vdots \\ \tau_M F^M \end{pmatrix}. \tag{5.13}$$

Optimality Conditions as Linear System For the state y as well as the adjoint state p, let us (for all time steps) collect the FE coefficients in $\overline{y} := (C_Y^m)_{m=0}^M$ and $\overline{p} := (C_P^m)_{m=0}^M$, respectively.
Then, we may compute $\overline{y}$ by a system of type (5.13) and $\overline{p}$ by a system similar to (5.13). (Note that the adjoint state equation for p is backward in time and a *final* value is imposed on p. Hence, slight modifications to (5.13) are necessary.) We may then use an iteration of "fixed point" type in order find a solution to the coupled system of these two equations.
Alternatively, we may again build an (even larger) block matrix in order to obtain a *single* linear system to solve. In particular, we denote the block matrix in (5.13) corresponding to the state by Y and the block matrix corresponding to the adjoint state by P (paying attention to the modifications necessary; see above). Introducing matrices Y_{con} and P_{con} to take care of the couplings of $\overline{y}$ and $\overline{p}$ (according to Corollary 5.2.5) as well as suitable RHS f_y and f_p, we obtain the linear system

$$\begin{pmatrix} Y & P_{\text{con}} \\ Y_{\text{con}} & P \end{pmatrix} \begin{pmatrix} \overline{y} \\ \overline{p} \end{pmatrix} = \begin{pmatrix} f_y \\ f_p \end{pmatrix}. \tag{5.14}$$

Feasibility Note that, again, the resulting system (5.14) easily becomes *huge*: The block matrix is of size $2qM \times 2qM$. Even for rather coarse discretizations with linear ansatz functions and a problem in two spatial dimensions, we easily have $q \approx 10^5$ and $t \approx 10^2$ – which yields a problem in approximately 10^7 variables. In Suboptimal Control, we shall in particular reduce the term q significantly to make such control problems more feasible.

5.4 POD Suboptimal Open-loop Control

In Chapter 4, we have constructed a POD reduced-order model for Evolution Problem 1.3.2. Let us now make use of this model in order to reduce the numerical costs of solving a respective Optimal Control problem. Since this procedure will in general not yield an "optimal control" of the full system, we refer to this approach as *Suboptimal Control*. Corresponding numerical examples may be found in Section 6.3.

Procedure We give a brief overview of different types of suboptimal control. We derive a "suboptimality system" and comment on respective error estimates. Then, we investigate the actual "benefit" on a discrete level as well as "numerical improvements". Let us finally consider the main obstacle in POD suboptimal control: ensuring the applicability of the POD Basis. We point out two ways of solution and comment on the respective treatment in a numerical context.

POD Suboptimal Control – Link to "Optimal Control" Since POD shall turn out to only present a special form of *discretization*, we may apply the theory developed on optimal control problems without severe changes. In particular, we shall obtain the same choices of procedure (such as choosing to "discretize" or to "optimize" first).

Therefore, we shall mainly concentrate on the major problem of POD suboptimal control: Due to the *dependence* of the POD Basis on the solution of a system, the basis depends on the "data" in a system and in particular, it depends on the *control variable*, which we aim to determine (refer to Subsection 5.4.5 for details).

5.4.1 Introduction to POD Suboptimal control

Let us briefly characterize the method of POD suboptimal control in order to understand that, from a theoretical point of view, this procedure may directly be deduced from the "general" approach, considered in Section 5.2.1. In practice however, the dependence of the model on the solution (i.e., the state) presents constraints on the optimality of the solution. In this way, we wish to actually motivate the term "suboptimal" control.

Reduced-order modeling as discretization In order to setup a POD reduced-order model, we choose the span of a POD Basis as a test space in a Galerkin ansatz for the system.
From a mathematical point of view, "reducing the model" therefore is just a special form of *discretization*. However, the approach significantly differs from "general" approaches (such as FE discretizations) since the POD Basis *depends* on the actual system of concern.

Sub-Optimality of POD Reduced-Order Control In some sense, *all* "discretizations" of an optimal control problem lead to a solution which is "suboptimal", yet it still makes sense to differentiate amongst different discretization schemes:
We refer to the solution to be suboptimal since our (POD) basis for the discretization is very "restrictive", i.e. the state is optimal in a rather confined manner (in particular, only the "characteristics" of the state modeled by the POD Basis matter).
In this sense, FE discretizations lead to a "more optimal" solution: The ansatz functions are quite general. Therefore, the optimal control is optimal in a "quite general" sense.
In other words, a POD "suboptimal" control minimizes a given functional only on the *modeled characteristics* of a corresponding state (instead of the state itself).

5.4.2 Types of Suboptimal Control Strategies

Let us briefly mention the basic types of sub-optimal control strategies. Reduced-order modeling being a form of "discretization", we expect to find approaches similar to the ones introduced in context of numerical treatment of Optimal Control problems (refer to Section 5.3).
If we additionally differentiate between the time- and the space discretization, we actually find another way of suboptimal control of space-time dependent systems: "instantaneous control". In the remainder we shall however focus on the POD reduced-order model approach.

Model Reduction vs Instantaneous Control Optimal control problems of time-dependent systems may be tackled "suboptimally" in two different ways. These ways (roughly speaking) originate from the two ways of discretizing non-stationary problems: "vertical" and "horizontal" method. (refer to Section 1.4)
Choosing the horizontal method, we discretize the system in time and obtain a sequence of stationary problems. We may then compute an optimal control for each *stationary* problem individually, which we then combine to one "time-global" optimal control. In other words, we calculate an optimal control which is optimal only for each time interval of the time discretization. We call this procedure *"instantaneous control"* (for more details, see Hinze and Kauffmann 1998, for instance).
On the other hand, we may make use of the vertical method, i.e., we discretize the PDE by (say) the FE method and obtain a system of ODEs (Proposition 1.4.2). We represent snapshots of the state by a POD Basis and setup a reduced-order model by choosing a corresponding ansatz space. We may then apply the theory of Section 5.2 in order to solve the resulting "suboptimal control via Model Reduction" problem.

Recalling the Ways of Numerical Treatment As mentioned above, in the process of optimization, POD is a "substitute" for FE discretization (in the Galerkin ansatz sense). Therefore, the choices of procedure "coincide" with the choices of Subsection 5.3 ("Discretize then Optimize" or vice versa).
"Discretizing" the control problem, in our case leads to a standard quadratic programming problem (Subsection 5.3.1). Using the POD Method (and a time discretization) we expect the problem to be way smaller than in an FE case say. Standard codes (like "quadprog") should be able to solve these problems more quickly. In the control-unconstrained case, the linear system (5.12) shall be reduced in size significantly. We shall however not touch upon this matter any further.
Alternatively, we may use the optimality conditions of Proposition 5.2.4. By means of the reduced order model, we may then reduce the size of the Evolution Problems involved in these conditions (refer to Subsection 5.4.3). Discretizing the system, we then obtain a way smaller linear system than (5.14) (if we assume to have no constraints on the control). Note that this procedure however is not as straightforward as it may seem (refer to Subsection 5.4.5).

5.4.3 Suboptimal Control Problem and Solution

In analogy to Subsection 5.2.3, let us introduce a *sub-optimality system*. Although this system will differ only slightly from the case of a "general" ansatz space, we wish to explicitly denote it in order to enlighten the discussion of the choice of snapshot sets in the subsection below. We close this subsection by a short comment on error estimates of POD suboptimal control.

Ingredients Instead of the full Evolution Problem (5.1), we shall use the reduced-order model of Problem 4.1.1. Then, it suffices to derive a respective model for the adjoint equation as well as the corresponding optimality condition.
Technically, reduced-order models are determined by their ansatz space. Let us denote the ansatz space corresponding to the state model by $\mathcal{V}_y^\ell$ and the one corresponding to the adjoint state model by $\mathcal{V}_p^\ell$. We think of these ansatz spaces to be the spans of the POD Basis which represent snapshots of the state and the adjoint state, respectively. (A priori, the reduced-order models for the state and the adjoint state are different.)

Sub-optimality System We have discussed Optimal Control problems and corresponding optimality conditions for "general" ansatz spaces in Subsections 5.2.1 and 5.2.3. Let us now gather all these findings for the case of POD ansatz spaces. (Analogously to the proof of Proposition 5.2.1, which assured a unique solution of the "general" problem, we also may obtain a result on the existence of a unique solution.)

Corollary 5.4.1 (Sub-Optimality system)
Let $\ell \in \mathbb{N}$ be fixed and J the target functional defined in (5.3). The pair $\bar{x}^\ell = (\bar{y}^\ell, \bar{u}^\ell)$ is the (unique) solution of problem

$$\min_{u \in \mathcal{U}_{ad}} \hat{J}^\ell := J(y^\ell(u), u) \tag{SROC}$$

if and only if $\bar{x}^\ell$ fulfills the state equation in $[0,T]$

$$\frac{d}{dt}\left(y^\ell(t), \psi\right)_H + a(y^\ell(t), \psi) = (f(t), \psi)_{V',V} \qquad \text{for all } \psi \in V_y^\ell, \tag{5.15a}$$

$$\left(y^\ell(0), \psi\right)_H = (y_0, \psi)_H \qquad \text{for all } \psi \in \mathcal{V}_y^\ell \tag{5.15b}$$

and with the unique Lagrange-multiplier $\bar{p}^\ell \in W(0,T)$ satisfies the following *adjoint*

equation in $[0,T]$

$$-\frac{d}{dt}\left(p^\ell(t),\psi\right)_H + a(p^\ell(t),\phi) = \alpha_1\left(z_1(t)-y^\ell(t),\psi\right)_H \qquad \text{for all } \psi\in\mathcal{V}_p^\ell \tag{5.16a}$$

$$\left(p^\ell(T),\psi\right)_H = \alpha_2\left(z_2-y^\ell(T),\psi\right)_H \qquad \text{for all } \psi\in\mathcal{V}_p^\ell \tag{5.16b}$$

as well as the corresponding *optimality condition*

$$\left(G^\ell(\bar{u}^\ell, u-\bar{u}^\ell\right)_{\mathcal{U}} \geq 0 \quad \text{for all } u\in\mathcal{U}_{ad}, \tag{5.17}$$

where we have defined the approximation $G^\ell : \mathcal{U}\to\mathcal{U}$ of the operator G by

$$G^\ell(u) := \sigma u - \mathcal{B}^* p^\ell. \tag{5.18}$$

Proof.
Essentially, we have to put together the state equation (5.1) as well as the optimality conditions of Proposition 5.2.4. In perfect analogy to the state equation, we may deduce a POD approximation for the adjoint state equation appearing. Technically, this formulation is obtained by substituting V with $\mathcal{V}^\ell$ and y with y^ℓ in (5.5).
Since $y^\ell, p^\ell \in W(0,T)$ are the unique solutions to (5.15) and (5.16), respectively, the operator G^ℓ is well-defined. □

Error Estimates for POD Optimization Since this issue is not the focus of this thesis, let us quote the "main result" of Hinze and Volkwein 2005: Let u denote the solution of the linear-quadratic optimal control problem and u^ℓ its POD approximation using POD Basis functions for the Galerkin ansatz. Then

$$\bar{u}^\ell - \bar{u} \sim \bar{p}^\ell - \bar{p},$$

where $\bar{p} = \bar{p}(u)$ and $\bar{p}^\ell = \bar{p}^\ell(u)$ denote the corresponding solutions of the continuous and discrete adjoint systems, respectively, and are associated to the same control u. (For details refer to Hinze and Volkwein 2005, Theorem 4.7.)
Note that this result was derived for the idealized situation that we assume to know the *exact* snapshots of the state which corresponds to the *optimal* solution. In this way, the problem of "non-modeled dynamics" (refer to Subsection 5.4.5) is overcome for the estimation.

5.4.4 Numerical Considerations

Since Suboptimal Control is only a special discrete variant of Optimal Control, the numerical treatment is essentially of the types considered in Section 5.3. Note however that in general a "sequence" of such POD suboptimal control problems has to be carried out (refer to Subsection 5.4.5).
Let us furthermore investigate where the actual "reduction" takes place and whether it suffices to use a *single* POD Basis (for the state as well as the adjoint state).

Resulting System in Unconstrained Case Let us shortly show which consequences Suboptimal Control yields on a "discrete level". In case that we did not impose boundary conditions, we ended up with a linear system of size $2qM \times 2qM$; for q degrees of freedom and M time steps (refer to (5.14)).
In a suboptimal context, the vectors $\overline{y}$ and $\overline{p}$ denote collections of coefficients w.r.t. a POD Basis, i.e. the system is reduced significantly: The block matrix is of size $2\ell M \times 2\ell M$. For (say) 10 POD modes and 100 time steps, this yields a system in 2000 variables (in contrast to 10^7 in the FE case of Subsection 5.3.3).

Reduction to One Ansatz Space? According to the optimality condition of Corollary 5.4.1, we have to setup two reduced-order models – namely one for the state y and one for the adjoint state p. Consequently, we have to deal with two snapshots sets and two resulting POD Basis. Additionally, we shall learn in the next subsection that it may be necessary to update the POD Basis regularly.
Let us therefore consider whether we can reduce the numerical effort by using only *one* POD Basis representing the state as well as the adjoint state. We may furthermore consider whether we should calculate this basis from snapshots of the state, the adjoint state or both the states.
In Hinze and Volkwein 2005, Section 5.2 a corresponding numerical study has been carried out. In particular, it was found that if snapshots of only one variable were included into the snapshots set, nearly no decay of eigenvalues for the other variable was achieved. (Recall that we generally expect this decay to be "exponential" – refer to Subsection 3.2.2.) Including both the variables achieved a satisfying decay.

Advantages vs Disadvantages In case the characteristics of both the states nearly coincide, it does not make sense to "maintain" two "similar" POD Basis. We may then represent the states by a "common" POD Basis.
Note however that, in general, the number of snapshots is increased and hence the calculation of a POD Basis becomes more expensive. Furthermore, a POD Basis essentially is the solution of an eigenvalue problem. Since the effort of the solution of such a problem is in general not linear in its size, we find that solving two "half-size problems" may actually be quicker than solving one "full-size problem". This is to say that computing two POD Basis from individual snapshot sets would be quicker than computing one POD Basis from a large set of snapshots.
Furthermore, note that if we include both the state variables into the snapshot set, we obtain a set of variables of potentially *"different scales"*. This may present a problem (refer to Subsection 4.3.2). In particular, *all* snapshots of one state might be of way lower energy than the snapshots of the other one. In this case, the POD Basis would only represent snapshots of the "energetic" state which would essentially lead to the same situation as including snapshots of only one state.

5.4.5 Tackling the Problem of "Non-Modeled Dynamics"

In this subsection, we introduce the problem of "non-modeled dynamics" and provide two solutions to it: an "adaptive POD algorithm" and "optimality system POD".
Let us refer to the state which corresponds to the optimal control of the system as "optimal state" and let us introduce the notion of an *"optimal"* POD Basis, i.e., a POD Basis which represents snapshots of the optimal state.

The Problem of "Non-Modeled Dynamics" A POD reduced-order model is based on POD modes which in turn are obtained from snapshots of the *state* of the system. In order to obtain the *state* of the system, we have to choose *all* data in the system, i.e., we have to provide "some" value for the control variable. If we now calculate an "optimal" control by means of the reduced-order model, there is no guarantee that this model is valid anymore.
In particular, we have "changed" the control, i.e., we have amended the data. Thus, the optimal state may have attained characteristics which were not present in the state corresponding to our initial "guess" of control. Consequently, these characteristics were not present in the snapshots represented by the POD Basis. Basing our reduced-order model on this POD Basis, we find that these characteristics (of the *optimal* state) are not "modeled" and thus, this optimal state cannot be found by our POD approach. We hence refer to this problem as the "problem of non-modeled dynamics".

Tackling the Problem of Non-Modeled Dynamics The POD reduced-order model is built on the characteristics of the state corresponding to an initial "guess" of the control.

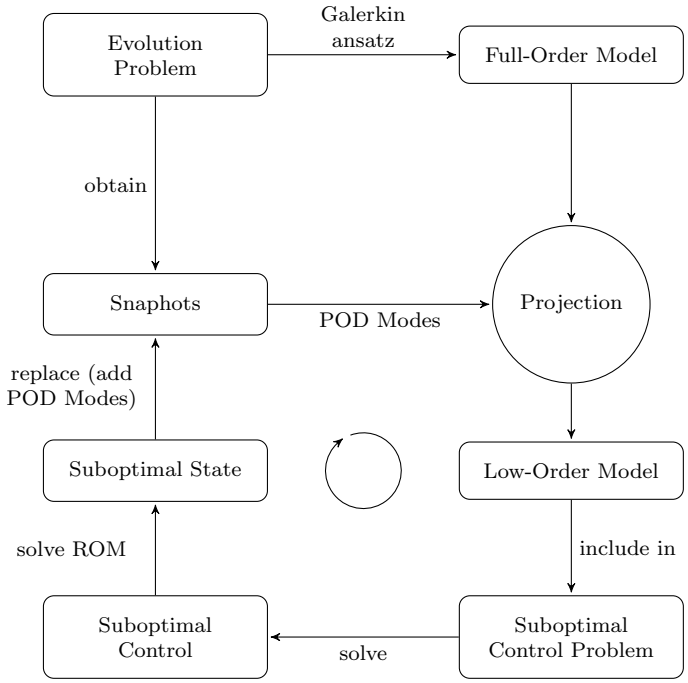

Figure 5.4: Visualization of the adaptive POD algorithm. (The upper two nodes are due to the initialization only – the actual iteration is marked by the circular arrow.)

An "optimal" control calculated on the basis of this model may in fact introduce characteristics into the state which have not been present in the previous state. Yet the control is only "suboptimal" in the sense that it minimizes a given functional together with only the *modeled characteristics* of the corresponding state. Therefore, we essentially have to make sure that our POD Basis contains "characteristics of optimal state".
Hence, one way to tackle the problem of non-modeled dynamics is to *update* the POD Basis as the control is changed ("adaptive POD algorithm"). Alternatively, we may include the choice of POD Basis into the optimality system ("OS-POD").

Adaptive Open Loop POD Control – Iteration with Update of the POD Basis As proposed above, we guess a control, take snapshots of the corresponding state and calculate a POD Basis. We then determine the corresponding POD suboptimal control as well as the (corresponding) suboptimal state.
Since our POD Basis does not necessarily contain the *characteristics* of this suboptimal state, we take snapshots from this state and *update* the POD Basis.
It is a matter of discussion whether to *add* the new snapshots to the snapshot set or to *replace* the snapshots: In the first case, the snapshot set increases with each cycle of the algorithm and hence the POD Basis becomes more expensive to calculate, whereas in the latter case, the POD Basis contains only information about the most recent suboptimal state.
A good compromise is to add the new snapshots to a set of all POD modes obtained so far: Since there are way less POD modes than snapshots, the snapshot set then increases slower in size but it still contains all the "essential" dynamics of suboptimal states calculated so far (since these dynamics are represented by the POD modes).
Let us now formally state this algorithm. Note that the procedure also is depicted in Figure 5.4. The algorithm is discussed in detail in Afanasiev 2002 (refer also to Hinze and Volkwein 2004, Section 4.1).

Algorithm 5.4.2 (Adaptive POD)
Choose a tolerance tol and an initial control estimate u_0. Compute snapshots by solving the state equation with $u := u_0$ and the adjoint equation with $y = y(u_0)$ and set $i := 0$.

S1: POD Determine ℓ as well as a POD Basis for $\mathcal{V}^\ell$ and construct a corresponding reduced-order model.

S2: Optimize Compute a solution u_i to the "sub-optimality system" of Corollary 5.4.1 (in one of the ways of Subsection 5.3).

S3: Termination or Snapshot Update If $\|u_i - u_{i-1}\| > \text{tol}$, compute snapshots by solving the state equation with control $u := u_i$ and adjoint equation with $y := y(u_i)$. Compose a new snapshot set according to the discussion above, set $i := i + 1$ and go back to S1.

Optimality System POD (OS-POD) The idea of this method is that we extend our optimality system (Corollary 5.4.1) by a "measure" for the POD modes to coincide with the "optimal" POD modes (see above).
The optimal solution then not only consist of the optimal control (with corresponding state) but also of the corresponding POD Basis: Essentially, we include the "optimization problem" to obtain an (optimal) POD Basis into the optimality system.
In this way, we simultaneously obtain the optimal control as well as the optimal POD Basis. We therefore circumvent the problem of "non-modeled dynamics" since the important dynamics (i.e. those of the optimal state) are modeled.
A drawback of this procedure is of course that we have *increased* the size of the optimality problem. But unfortunately, a thorough discussion of the method is beyond the scope of this thesis. Note however that "OS-POD" was introduced in Kunisch and Volkwein 2006 in which also numerical examples are presented.

5.5 Outlook: Closed Loop (Feedback) Control

Let us briefly introduce the matter of *feedback control* as this presents a typical application of sub-optimal control strategies. (A thorough discussion unfortunately is beyond the scope of this thesis.) Most parts of the theory may be found in Volkwein 2006, Benner, Goerner, and Saak 2006 as well as Lions 1971. It turns out that the "linear quadratic problem" will in the discrete case be solved by the solution to a so-called *"matrix Riccati equation"*. This result may be established by "Hamilton-Jacobi" theory or by rearranging the theory for open-loop control (Subsection 5.2.3) and introducing suitable operators (refer to Subsection 5.5.2).

Introduction – Motivation Up to now we have considered open-loop control, i.e., we were given a control problem (as described in Subsection 5.2.1) and wanted to find an optimal control $\bar{u}$.
In practice, the following situation is much more likely: We measure the state of some "system" and wish to change it to another one – or alternatively wish to retain it (the temperature in a room for example). For that purpose, we wish to calculate a control *in terms* of the measurement.
It is especially in these sort of situations that a quick reaction is of greater importance than an exact solution. Hence, suboptimal control strategies are especially useful in these kind of circumstances; which is the main motivation of presenting the issue here.

Suboptimal Closed Loop Control Of course, we could also use suboptimal control strategies for tackling closed loop control problems. As explained in Section 5.1, in terms of application, this would be even more fruitful. But yet again, a thorough discussion would be far beyond the scope of this

thesis. Therefore, let us only mention that POD feedback control is presented in Kunisch, Volkwein, and Xie 2004 and that in general, there are two possible procedures (in analogy to open-loop control). There is the "approximate then design approach", which basically says that we first find a low-order approximation of a system for which we then may construct a controller. On the other hand, we may alternatively first determine a controller "on a continuous level" and then approximate it by low-order models. This method bears the advantage of including more "physical details" into the controller (according to Atwell and King 1998).

5.5.1 The Linear-quadratic Regulator Problem

Let us explain the type of feedback-control problem which we shall consider and let us outline the general procedure. This section being an "outlook", we immediately consider space-discretized PDE, i.e., systems of ODEs.

Summary of Procedure In the language of Subsection 5.3, we proceed in the manner "semi-discretize – optimize – semi-discretize". We semi-discretize the parabolic IVP (refer to Problem 1.3.5) in space, by means of an FE method say, and obtain a system of ODEs. We then "optimize" the system – in the sense that we setup a suitable "LQR problem" and solve this for the "corresponding controller". Inserting the controller into the ODE System, we retrieve the *"closed-loop system"*.
We then discretize the closed-loop system in time in order obtain a solution, the *"optimal trajectory"*.
By means of the controller, we may then compute the *optimal control*.

LQR Controller Given an objective functional and an ODE constraint, the aim of feedback control is to calculate a *controller*, i.e., a function which maps a state measurement to an optimal control (in terms of the objective).
In particular, we consider linear-quadratic problems. The feedback control problem is then known as *linear-quadratic regulator problem* (LQR). (There are variants like *linear quadratic Gaussian* (LQG), which only involve a state "estimate" and hence noise and disturbances may be added.)
As indicated already, there are two possibilities in terms of the objective: "stabilization of the state" or "tracking a certain state".

LQR Problem Statement Let us now state the issues of the previous subsection in mathematical terms: We assume that there are m_u degrees of freedom in the control and m_x degrees of freedom in the sate. Let $Q, M \in R^{m_x \times m_x}$ be symmetric and positive semi-definite and let $R \in R^{m_u \times m_u}$ be symmetric and positive definite. Choose $A \in R^{m_x \times m_x}$, $B \in R^{m_x \times m_u}$ and $x_0 \in R^{m_x}$. The final time T shall be fixed, but the final state $x(T)$ shall be free. We may now define the actual problem:

Problem 5.5.1 (LQR Problem)
Find a *state-feedback control law* of the form

$$u(t) = -Kx(t) \quad \text{for } t \in [0,T]$$

with $u : [0,T] \to \mathbb{R}^{m_x}$, $x : [0,T] \to \mathbb{R}^{m_x}$, $K \in R^{m_u \times m_x}$ so that u minimizes the *quadratic cost functional*

$$J(x,u) = \int_0^T x(t)^T Q x(t) + u(t)^T R u(t) dt + x(T)^T M x(T),$$

where the state x and the control u are related by the *linear initial value problem*

$$x(t) = Ax(t) + Bu(t) \quad \text{for } t \in (0,T] \qquad \text{and} \qquad x(0) = x_0.$$

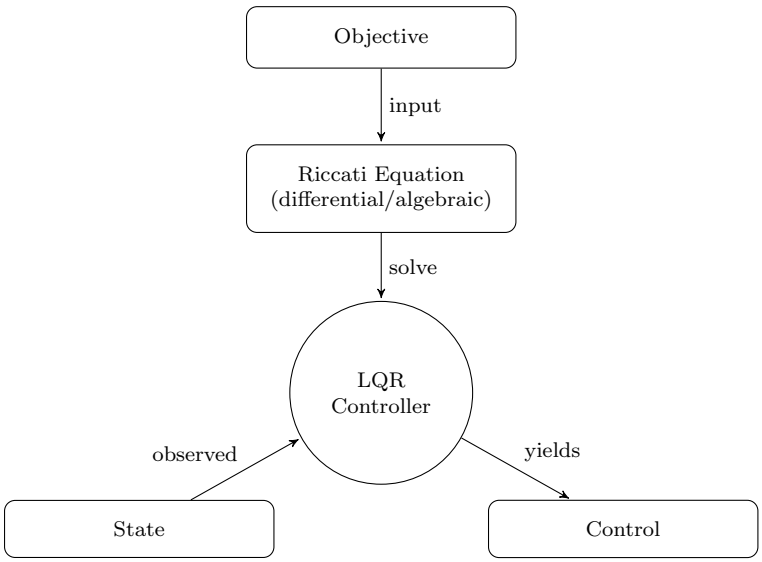

Figure 5.5: Schematic diagram of an LQR Controller and its calculation based on a given objective.

"Stabilizing" vs "Tracking" Interpreting Problem 5.5.1, we aim to "steer" the state of the system to the state $x = 0$ as good as possible. The terms $x(t)^T Qx(t)$ and $x(T)^T Mx(T)$ are measures for the "state accuracy" and the term $u(t)^T Ru(t)$ measures the "control effort".
For this type of "tracking control", there is a "common trick" to transform it into a "stabilizing" statement – refer to Benner, Goerner, and Saak 2006, Subsection 2.2 for example.

5.5.2 Optimality Conditions for the LQR Problem

In order to characterize a solution of the LQR Problem 5.5.1, we could rearrange the statement of the optimality conditions for open-loop control and establish an operator that maps an observation of the state to an optimal control.
Alternatively, we may use the *Hamilton-Jacobi-Bellman-Theory* (HJB theory), which is even applicable to non-linear problems. (For details on this theory, refer to Volkwein 2006, Section 3.3 or Lions 1971, Subsection III.4.7).
Both the approaches lead to a so-called *"Riccati equation"*. This way of calculating a feedback controller as well as its "role" are depicted in Figure 5.5.

The LQR Case Let us quote the following result which teaches us that the LQR controller is given by the solution to a *"matrix Riccati equation"*.

Proposition 5.5.2 (Construction of an LQR Controller)
Let $P : [0, T] \to \mathbb{R}^{m_x}$ be the solution to the *matrix Riccati equation*

$$-P'(t) = A^T P(t) + P(t)A + Q - P(t)BR^{-1}B^T P(t), \quad t \in [0, T), \tag{5.19a}$$
$$P(T) = M. \tag{5.19b}$$

Then, the optimal state-feedback control to Problem 5.5.1 is given by

$$u^*(t) = \big(-R^{-1}B^T P(t)\big)x(t).$$

Proof.
A derivation via the Hamilton-Jacobi-Bellman equation (HJB) might be found in Volkwein 2006, Section 3.3 or Dorato, Abdallah, and Cerone 1995. □

Connection to Open-loop Case and Affine Feedback Control Let us point out the rather close connection of the theory of open- and closed-loop-control in this context. (If not mentioned otherwise, all references in this paragraph refer to Lions 1971, Chapter III.)
We take the solution of an unconstrained *open*-loop control problem which satisfies a system of two coupled Evolution Problems (refer to (2.24) or in this thesis: Problem 5.2.5).
Consider this system with "initial time" s and "initial value" $h \in H$. For convenience, let us retain the names of variables, i.e., we set $y(s) := h$ ((4.12) and (4.13) with some changes in the notation). The second component of the solution $(\bar{y}, \bar{p})$ of this system induces a continuous, affine mapping $F(s) : H \to H,\ h \mapsto \bar{p}(s)$ for each s allowed (Corollary 4.1). According to Corollary 4.2 and Lemma 4.3, we may state this mapping in form of an *"affine"* equation:

$$p(t) = P(t)y(t) + r(t), \quad P(t) \in \mathcal{L}(H, H), \quad r(s) \in H.$$

The mapping P is given by a *matrix Riccati equation* ((5.19) in this thesis; or (4.55) and (4.57)) and r is given by an *abstract parabolic equation* (refer to (4.56) and (4.58)).

Further Variants Of course, there are further types of problems: We could think of an *infinite time horizon*, for instance. In this case we would obtain an *algebraic* Riccati equation to solve. (Refer to Lions 1971, Section III.6, in particular to (6.22); again for the "affine ansatz".)
Furthermore, one might like to impose constraints on the control, for example. For that purpose, "dynamic programming" via the "Bellman principle" is of help (Lions 1971, Section 4.8).

Numerical Treatment A lot of theory has been developed on how to solve equations of "Riccati type". (Refer for example to the talk Benner 2001 or to the reference Benner, Goerner, and Saak 2006 in which the authors also comment on actual implementations.) Let us also mention that a "discrete approach" to a feedback control problem is presented in Lions 1971, Sections III.4 and III.5.

Chapter 6

Numerical Investigations

In this chapter, we mainly wish to illustrate the theory developed for the POD Method. We present many simple examples in order to aid understanding the *characteristics* of the method, such as its asymmetry in space and time. Furthermore, we investigate aspects such as the mean reduction, which have not been touched upon in greater details so far. Also the influence of the snapshot error on the POD Basis shall be studied. Finally, we provide results for an "academic" optimal control problem of the non-stationary Heat Equation.

Procedure Analogously to the theory, we first of all apply the POD Method to ensembles that do not necessarily come from the solution of an Evolution Problem in order to investigate the method itself. Then, we shall think of these ensembles to be taken from a solution of an IVP for the non-stationary Heat Equation and therewith set up reduced-order models for this equation. Finally, we consider two examples of Suboptimal Control.

Role of POD – "*Snapshot* Galerkin Approach" We should carefully investigate the actual role of POD. Suppose we are given a set of snapshots. The role of POD is now to represent them well. In case they are perfectly different, POD just shall not manage to reduce the order of a descent approximation.
Yet the FE-model might still be "reduced" by projecting it on the space of all snapshots. In this case we so to say have used a "*Snapshot* Galerkin Approach" instead of a "POD Galerkin Approach".

6.1 Numerical Experiments for POD on Discrete Ensembles

In this section, we are only interested in the *actual representation* of "discrete ensembles", i.e., we do not care about the origin of the elements of the ensemble. In other words, by proceeding in this way, we do not have to deal with the (additional) error of solving a resulting low-order system in order to judge on the quality of the approximation of a reference solution of an IVP. Instead, we may concentrate on the effect of reducing the system to a low-order rank.
Furthermore, we may freely choose the ensemble to be approximated, i.e., we could even use ensembles, which are *not* generated by functions which are a solution of an Evolution System. (Hence, our ensemble does not have to be a "snapshot set".)

Relation to Other Sections Basically, we let the ensemble be generated by certain functions Z, depending on two parameters which do not have anything to do with any particular Evolution System necessarily. (In terms of Section A.2 we could think of Z as a "signal".)
In Section 6.2, we shall *construct* parabolic IVPs such that those functions Z denote a reference solution. In this case, the "discrete ensemble" actually becomes a *snapshot set* of the respective IVP.

Consequently, at this stage the "low dimensional approximation" cannot be obtained from a ROM (since there is no model to reduce). Thus, we shall obtain the approximation in the fashion of *"truncating a signal"* (see Subsection A.2.2), i.e., by taking advantage of a "bi-orthogonal" decomposition.

6.1.1 Discrete Ensembles and their Low-rank Approximation

Let us define some notation for calculating approximations without the IVP "background".

The Discrete Ensemble We let the ensemble members depend on two parameters, which we call "space" and "time" (since they in terms of POD admit the roles, which space and time would for POD on snapshot sets).
To setup the *discrete* ensemble, we shall use an equidistant grid Ω_h with mesh size h for the space dimension and an equidistant grid Γ_τ for the time dimension (time step size τ). For $x_0 < x_1$, $t_0 < T \in \mathbb{R}$ as well as $q = \frac{x_1 - x_0}{h} + 1$ and $n = \frac{T - t_0}{\tau} + 1$ we introduce

$$x_i = ih + x_0, \quad i = 0, \dots, q-1 \qquad \text{and} \qquad t_k = k\tau + t_0, \quad k = 0, \dots, n-1.$$

We assume that all data is chosen in such a way that there holds $q, n \in \mathbb{N}$. If not mentioned otherwise, we shall assume $t_0 = 0$.
In order to improve readability, let us compactly denote those grids by means of the following notation

$$\Omega_h = [x_0 : h : x_1] \quad \text{and} \quad \Gamma_\tau = [t_0 : \tau : T].$$

We set up a "discrete surface" which is parameterized by two coordinates and hence said to be two dimensional. We call it "Z" if it is parameterized by a function:

$$Z(\Omega_h, \Gamma_\tau), \quad Z : \mathbb{R} \times \mathbb{R} \to \mathbb{R}.$$

We then choose our ensemble parameter space to be Γ_τ and the *discrete ensemble* to be

$$\mathcal{V}_P = y(\Gamma_\tau), \quad y : \mathbb{R} \to \mathbb{R}^q, \quad y(t) := Z(\Omega_h, t) \quad \text{for } t \in \mathbb{R}.$$

In terms of Evolution Systems, this would either correspond to an FE vector or an FE solution if the space is of one dimension.
Note that since we do not assume the data to origin from a dynamical system and furthermore use equidistant grids, we may apply the POD Method with D and W the identity, respectively.

Obtaining the Low-rank Approximation – Bi-Orthogonal Decomposition We consider a "low rank approximation" of a *two-dimensional* surface Z.
In particular, we setup a POD Basis based on the deduced ensemble $\mathcal{V}_P$ which captures the essential "information" in the surface.
In the context of ROM, we would project the *Evolution System* onto the span of the POD Basis in order to obtain a low order model. We would then solve this low-order problem to obtain the coefficients of the POD modes in order to build a low-order solution from the POD Basis.
As we wish to exclude the error of solving this low-order system and concentrate on the error induced by the POD Approximation, we determine those coefficients *directly*. For that purpose, we could either project the *surface* onto the span of the POD Basis or could use a *bi-orthogonal* decomposition of the signal. This yields the POD modes as well as the respective coefficients (see Subsection A.2.2).

Remark 6.1.1 (Actual Calculation – Spectral Cut-off)
In our discrete context, we may denote the ensemble $\mathcal{V}_P$ as an Ensemble Matrix Y (see Definition 3.1.4). We may then determine a POD Basis by calculating an SVD $Y = U\Sigma V$ (according to Theorem 3.1.5). The "bi-orthogonal decomposition" of the ensemble is then given by the columns of U (POD modes) and the columns of V. Their contribution decreases since the singular values in Σ decrease.

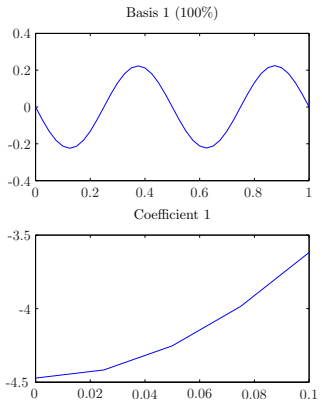

Figure 6.1: POD Basis obtained from the time interval $[0, 0.2]$. Only the dominating component is extracted by the POD Method.

In other words, we may obtain the *low-order approximation* of the surface S by neglecting modes with low singular values such that the remainder gathers sufficient "energy". In numerical mathematics this is well-known as "*spectral cut-off*", which in "Matlab notation" reads for the order k:

$$Z_k := U(:, 1:k) * \Sigma(1:k, 1:k) * V(:, 1:k)'.$$

6.1.2 Energy of Modes – POD vs Fourier Decomposition

According to Proposition A.1.4, the POD Method presents a generalization of the Fourier decomposition which we in this context shall understand as a "decomposition into trigonometric modes of different frequencies". Let us investigate this relationship as well as the notion of energy by means of two examples which are made up of Fourier modes.

Problem: Two Fourier Modes of different "intensity" We wish to illustrate the "energy" which is captured in the POD modes. For that purpose, we consider a surface parameterization made up of two components whose "contribution" varies over time. By varying the time interval we expect the POD Method to fail to extract both the components for some choices. In particular, we consider an ensemble generated by

$$Z(x,t) = \cos(2t\pi)\sin(4\pi x) + t^2\sin(2\pi x), \quad x \in [0:0.025:1], \quad t \in [0:0.025:T],\ T \in \{0.1, 1.25, 2\}.$$

Results Applying the POD Method with the aim to capture 0.99% of the energy, we find for

$T = 0.2$ that the first component dominates the overall energy contribution, and the POD Basis consists of just this element. The second component obviously makes up less than 1% of the energy in the ensemble.

$T = 1.25$ that two components are captured, but these are not equivalent to the actual Fourier modes (yet of course linearly depended on them). Thus, we see that the POD Method does not necessarily yield Fourier modes, even when applied to an ensemble made up of these.

$T = 2$ that the second component dominates, yet both components are captured and at least "look" similar to Fourier modes.

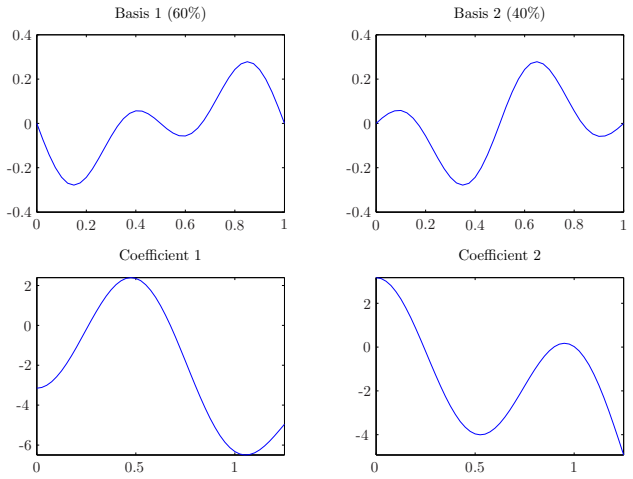

Figure 6.2: Time interval $[0, 1.25]$: Two non-Fourier modes are captured.

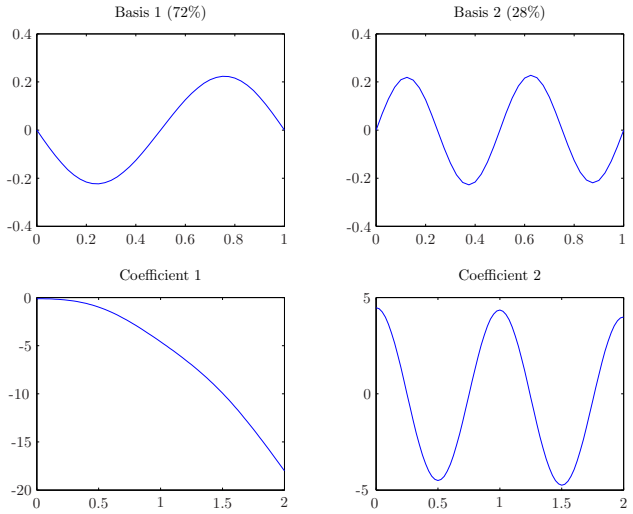

Figure 6.3: Time interval $[0, 2]$: Both the Fourier modes are captured.

These results are depicted in the Figures 6.1 to 6.3, respectively. Their energy contribution to the ensemble is shown in the respective title.

Note that the "domination" of a mode is reflected in the magnitude of the respective *coefficient* as the modes, of course, are *orthonormal*.

Problem: "Many" Fourier Modes Let us now consider an example consisting of "many" Fourier modes, i.e., the frequency shall be varied with time (and thus, every ensemble member has got a slightly different frequency). Let the ensemble consist of 41 elements, defined by

$$Z(x,t) = \sin(tx\pi), \quad x \in [-1 : 0.025 : 1], \quad t \in [0 : 0.1 : 4].$$

We find that 5 modes are needed to capture 99% of the energy in the ensemble of 41 frequencies, which are depicted in Figure 6.4.

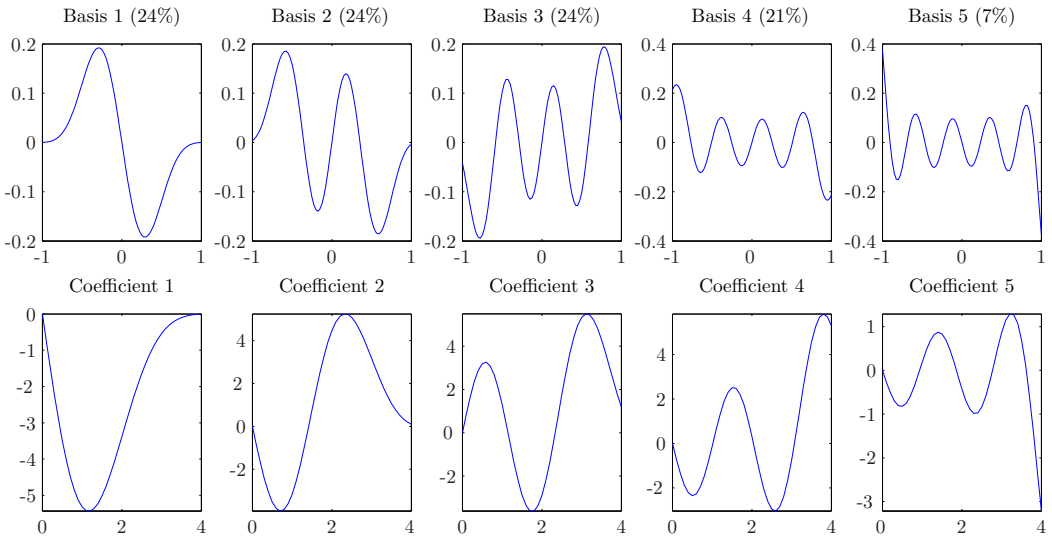

Figure 6.4: Modes approximating $\sin(txπ)$ (capturing 99% of energy).

6.1.3 Challenging Case

Of course, the POD Method might fail in certain situations. We give a simple example in which the POD Basis only yields a poor approximation of the ensemble of choice. This matter is also touched upon in the general discussion of the method in Section 4.3.

Description of Example We try to approximate

$$Z_a(x,t) = \exp(-a(x-t)^2), \quad a \in \mathbb{R}^+, \quad x \in [-2 : \frac{4}{99} : 2], \quad t \in [-2 : \frac{4}{99} : 2].$$

In particular this function equals 1 for $x = t$ and decreases very quickly to 0 in all other cases. The velocity of the decrease is controlled by the parameter a.

Conclusions With increasing a, the "information" travels more quickly. Hence the correlation in space (over ensemble members) decreases and the representation by the POD Basis becomes worse. This is reflected in Figure 6.5. (The relative error increases and the rank needed to capture a certain amount of energy in the ensemble increases as well.
To get an idea of how different the approximation actually looks for small ranks, we have depicted the first three approximations in Figure 6.6.

6.1.4 Study of Mean Subtraction

In Subsection 3.2.1, it was proposed that subtracting the mean from the snapshot set could yield better approximation results. (In the language of fluid dynamics, we so to say only model the "fluctuations" of the ensemble members.)

Realization For investigating the influence of the mean, let us make use of the Fourier example encountered before and shift it by a parameter $a \in \mathbb{R}$:

$$Z_a(x,t) = \sin(xt) + a, \quad x \in [-2 : \frac{1}{39} : 2], \quad t \in [-2 : \frac{1}{39} : 2].$$

Due to the symmetry of the sin-function on the interval of choice, we invoke that the mean of Z_a is equal to a. We now investigate the influence of the mean a on the POD Basis. In other words, we

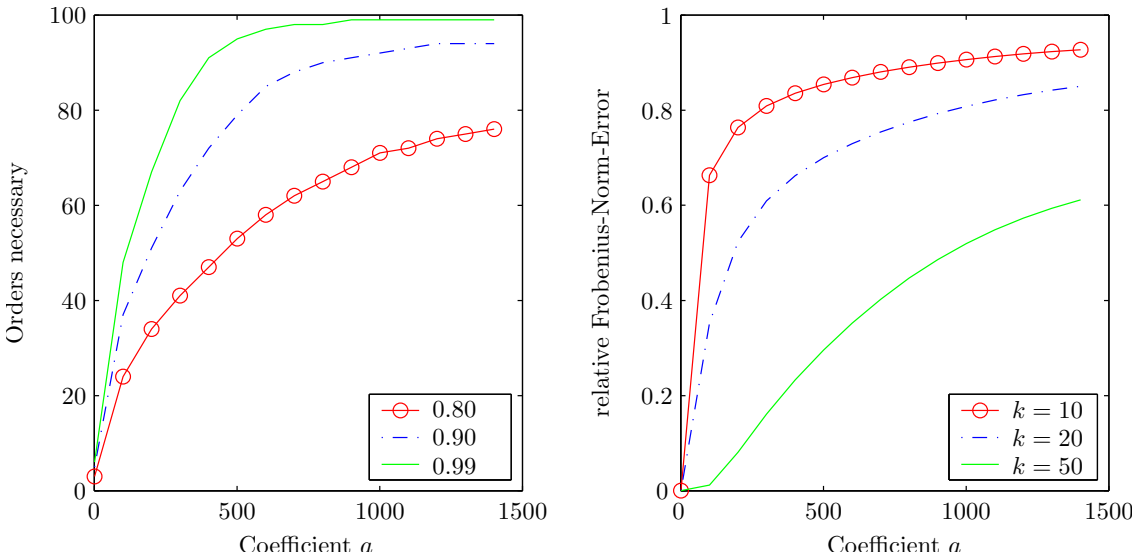

Figure 6.5: Approximation of $Z_a(x,t) = \exp(-a(x-t)^2)$ for different values of $a \in \mathbb{R}^+$: Orders necessary to capture desired amount of energy and relative error based on order of approximation.

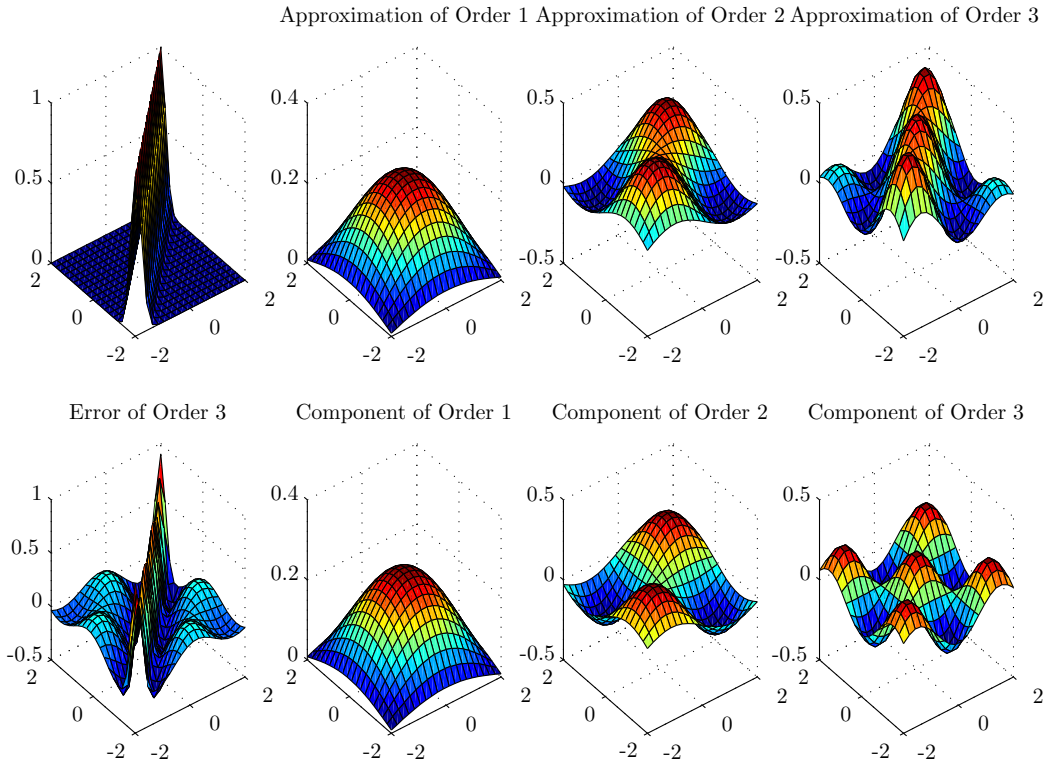

Figure 6.6: Actual surface, its first three approximations, the respective components as well as the error of approximation. (All plots are depicted on a coarser grid than calculated on.)

Mean	Energy 1	Energy 2	Energy 3	Energy 4	ℓ (99%)
0	23.6	9.33	0.612	0.0158	3
0.01	23.6	9.33	0.612	0.4	4
0.1	23.6	9.33	4	0.612	4
0.5	23.6	20	9.33	0.612	4
1	40	23.6	9.33	0.612	3
1000	40000	23.6	9.33	0.612	1

Table 6.1: Absolute energies of the first four POD modes, computed without "subtraction of mean".

Mean	Energy 1	Energy 2	Energy 3	Energy 4	ℓ (99%)
0	0.703	0.278	0.0182	0.000471	3
0.01	0.695	0.275	0.018	0.0118	4
0.1	0.629	0.248	0.106	0.0163	4
0.5	0.441	0.373	0.174	0.0114	4
1	0.544	0.321	0.127	0.00832	3
1000	0.999	0.00059	0.000233	0.0000153	1

Table 6.2: Relative energy contributions of the first four POD modes, computed without "subtraction of mean".

wish to find out what the improvement could be when subtracting the mean and hence applying the POD Method for the case $a = 0$.

Numerical Results Looking at Table 6.1, we see that in the absolute energy of the mean increases as a increases, yet all other energy components remain unchanged. Of course, the mean gains "relative" importance and eventually dominates all other components (as we see in Table 6.2).
If we subtract the mean from the ensemble members, we obtain the energy contribution of the case $a = 0$ for all choices of a.
In Figure 6.7, we again see that with increasing mean the basis function representing the mean gains importance in the representation, dominating all "fluctuations" for $a = 1000$.

Advantage 1: Further Reduction of Rank In our example, we have seen that the mean simply presents the major POD mode. By subtracting it beforehand, we may thus further reduce the cardinality of the POD Basis, which in turn further would reduce a reduced-order model say. (Since a typical cardinality of such a model is 10, this reduction is at least considerable.)

Advantage 2: Influence on Scaling and Stability We have seen in the example above that if the mean contribution becomes comparably large, the energy contribution of the "fluctuations" is decreased, which leads to many small eigenvalues. This in turn may lead to numerical instabilities in terms of determining the number of modes ℓ to use. In particular, due to the "down-scaling" of the eigenvalues, the *number* of POD modes to be used could critically depend on the amount of energy to be captured. Furthermore, in an extreme case (such as $a = 1000$), the mean may capture more than 99% of the energy and hence the POD Method only yields the mean as a basis function, which surely is not desired.

6.2 Numerical Examples for POD-ROM

Having investigated the POD Method on "discrete ensembles", we now wish to actually use it to construct low-order models for parabolic IVPs, in particular for the non-stationary heat equation.
In other words, we let our discrete ensemble be generated by a solution to the respective IVP. Hence, we may speak of the ensemble to be a *snapshot set*.

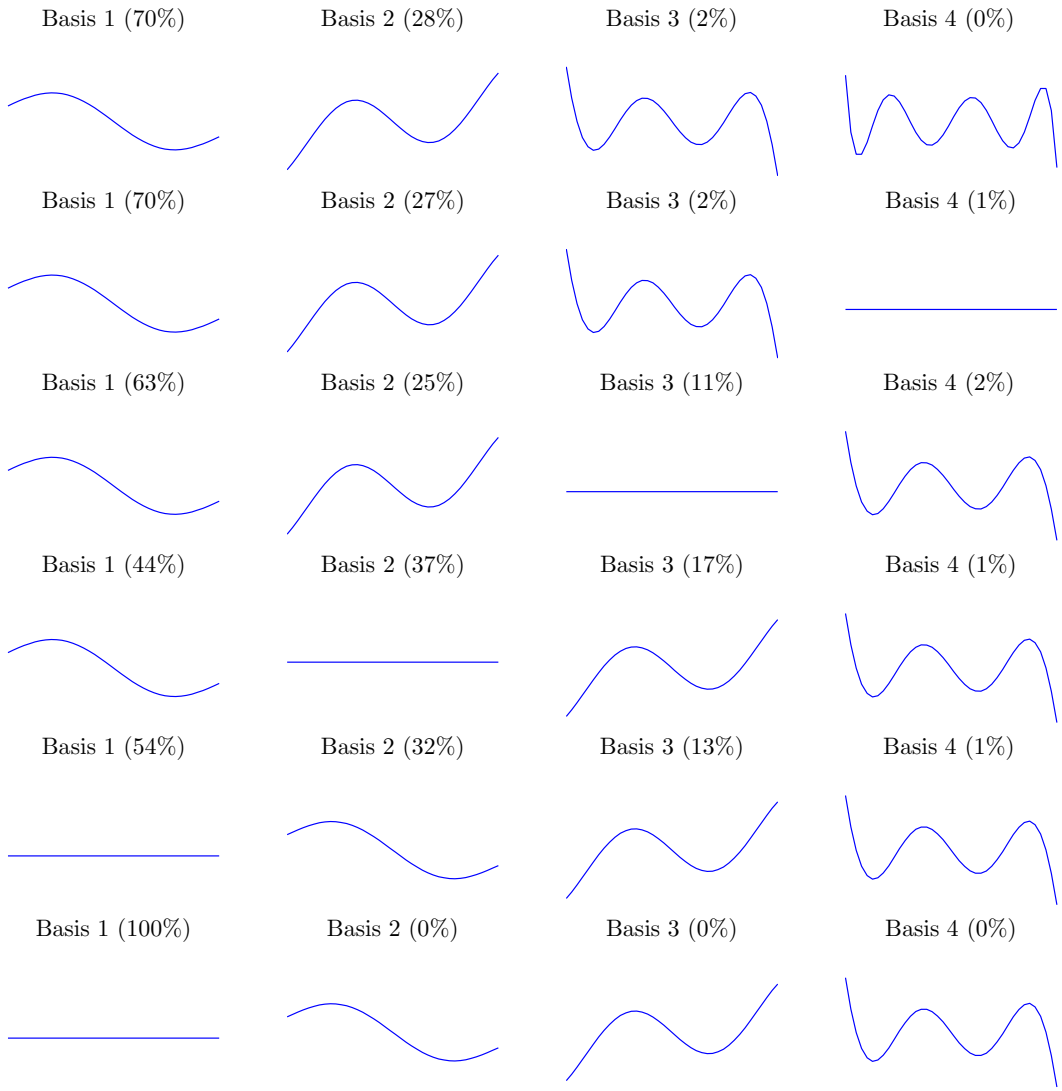

Figure 6.7: Investigating $Z_a(x,t) = \sin(xt) + a$. POD modes for (row-wise) $a \in \{0, 0.01, 0.1, 0.5, 1, 1000\}$. (The ranges are $(-2, 2)$ for the x- and $(-0.5, 0.5)$ for the y-axis.)

Note that throughout, we assume that the snapshots are given *exactly*. For that reason, we conclude the section by studying the dependence of the POD Basis on the discretization of snapshots. (Refer to Subsection 3.2.4 for the "theoretical" equivalent.)

The Snapshot Ensemble We consider a set $\mathcal{V}_P$ of n vectors $y \in \mathbb{R}^q$. In terms of discretization of Evolution Systems, we may generally think of these to be FE vectors of the solution at n time instances. Yet in later sections, we shall investigate one-dimensional problems in space and apply an FE scheme such that the vectors $y \in \mathcal{V}_P$ consist of the value of the solution at certain grid points in $\Omega_h \times \Gamma_\tau$.

Relation to Other Sections In Section 6.1, we have constructed examples in order to investigate the POD Method itself. Yet we have not considered the benefit in a *practical* context.
For instance, if we simply think of the ensembles used to be generated by a solution to an IVP, we assume to *know* the solution of the IVP. On the other hand, we wish to solve the IVP since we do

not know its solution.

Possible Scenarios of Applying the POD Method For simply *solving* an IVP, the POD Method is not of help necessarily since "some sort of solution" has to be available in order to obtain snapshots. Anyhow, there are three possible scenarios in which the POD Method might decrease computation time significantly:

1. We have to solve a collection of IVPs with slightly different data. (This usually is the situation in Optimal Control of IVPs.)
2. The solution is available on a coarse time grid and we wish to *interpolate* it on a fine grid.
3. The solution is available over a short period of time and we wish to *extrapolate* it to a longer time frame.

In the first case, we obtain the solution for one set of data, build the ROM and use this model for the other data, too. In total, this should be way quicker than solving the full problem for all the data. Of course, this procedure only works out fine if the "characteristics" of the solution are not changed too much with the data varying.
In the second and third case, we of course also gain a reduction in numerical effort. The quality of the low-order solution depends on the amount of "characteristics" of the solution on the fine grid (longer time frame) which is present on the coarse grid (short time frame) already.

6.2.1 Statement of Problem

Let us state all ingredients for the problem which we shall investigate reduced-order models for.

The Non-Stationary Heat Equation With the notation of Subsection 1.3.3, the IVP for the Instationary Heat Equation with homogeneous Dirichlet boundary conditions for a suitable RHS f reads

$$\begin{aligned} \frac{\partial}{\partial t} y - \Delta y &= f \quad \text{in } Q, \\ y &= 0 \quad \text{in } \Sigma, \\ y(0) &= 0 \quad \text{in } \Omega. \end{aligned}$$

Due to Subsection 1.3.3, the Heat Equation is a special case of parabolic IVP and hence, we may apply all the theory on POD and ROM.

Numerical Approach We consider the Heat Equation in a *one*-dimensional spatial domain $\Omega \in \mathbb{R}$. Then the plot of the solution, depending on time and space, becomes a two-dimensional surface. This of course is convenient for the presentation.
We use a Finite Element method for the discretization in space and an implicit Euler scheme in time.

Construction of Data – "Challenging Example" We wish to choose the data such that the *parameterization* of the ensemble becomes the reference solution of our system. For instance, for the "challenging example" this reads:

$$\begin{aligned} f(x,t) &:= \left(2k(x-t) - 4k^2(x-t)^2 + 2k\right) \exp(-k(x-t)^2), \\ y_0(x) &:= \exp(-kx^2), \quad \text{reference}(x) := \exp(-k(x-t)^2). \end{aligned}$$

6.2.2 Extrapolating a Solution

We wish to "extend" the time frame of a previously calculated solution.

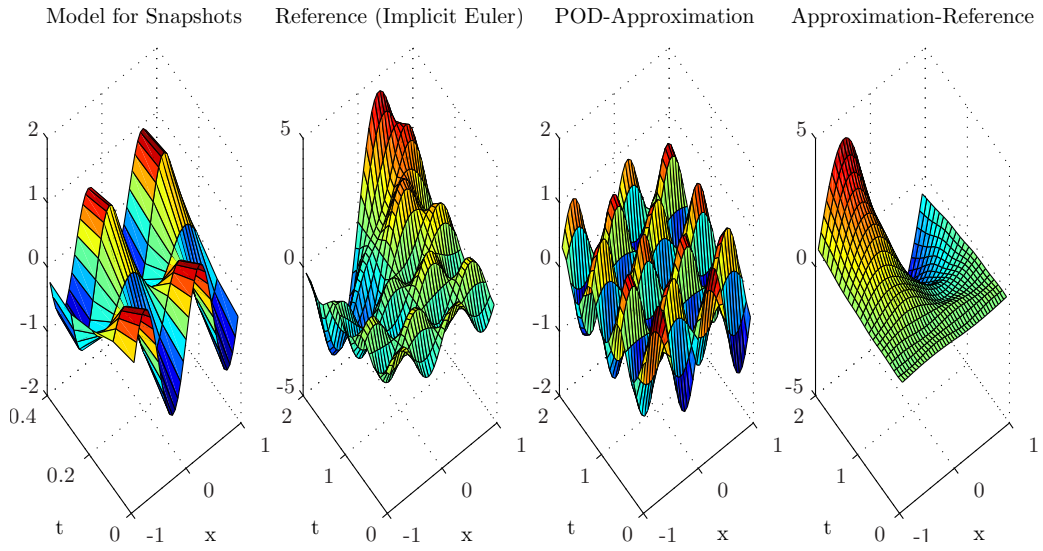

Figure 6.8: POD fails in "extrapolation": The error ("Approximation – Reference") grows with time.

Procedure To investigate this issue, we take snapshots at a rather short interval of time and "hope" that all characteristics of the solution are "coded" in the POD Basis. Then, we solve the resulting reduced-order order model on the desired period of time.

Example We have seen already in an example in Subsection 6.1.2 that POD might fail to extract features that are not "prominent" enough in the snapshots (see Figure 6.1).
In this example, the non-extracted component actually gains importance with time. Therefore, the reference solution is less matched by the POD low-order solution as time proceeds.
The snapshot model, the reference solution, the POD approximation as well as the error ("Approximation – Reference") are depicted in Figure 6.8.

Conclusion Again, we are left with the problem that it is hard to say whether this procedure shall work out fine. It essentially depends on whether the characteristics of the "solution to be extrapolated" are already present in the solution provided.
Additional theoretical comment: In this context, σ_n comes into play in the analytical error estimates of Chapter 4 (since the largest value of $t_{\bar{k}}$ used for all time instances in the solution, which are larger).

6.2.3 Interpolating a Solution

As mentioned above, one use of ROM in modeling could be: Obtain the solution to an IVP from on a *fine* grid by computing it on a *coarse* one and then *interpolating* it by means of the ROM.

Procedure We take (exact) snapshots on a coarse grid, calculate a POD Basis and solve the resulting ROM on a fine grid (10 times more grid points). We consider the "challenging example" of the previous section and show that this may workout quite well for "nice cases", but becomes "unusable" for others. Let t_{snap} denote the values of time, the snapshots are taken at. We choose

$$t \in [0 : 0.03 : 0.9] \quad \text{and} \quad t_{\text{snap}} \in [0 : 0.3 : 0.9].$$

Investigation for Slow Decrease If we choose the parameter a rather small, the POD representation works out quite fine (refer to Figure 6.9 for the case $a := 5$). We chose:

$$Z_a(x,t) = \exp(-a(x-t)^2), \quad a = 5, \quad x \in [-2 : 0.033 : 3].$$

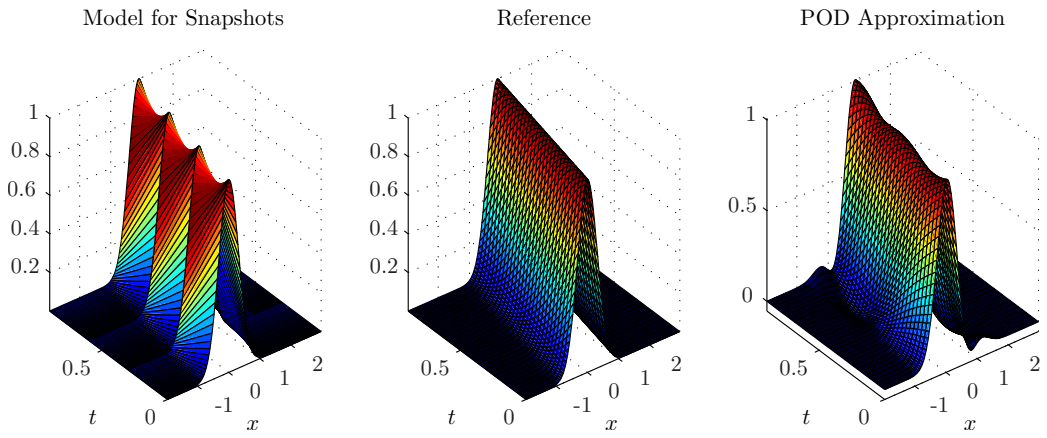

Figure 6.9: Interpolation of the "challenging example" for $a = 5$ – which behaves quite nicely. Snapshots are taken on a coarse grid. The low-order solution ("POD Approximation") is computed on the (fine) grid of the reference solution, capturing 99% of the energy in the snapshots.

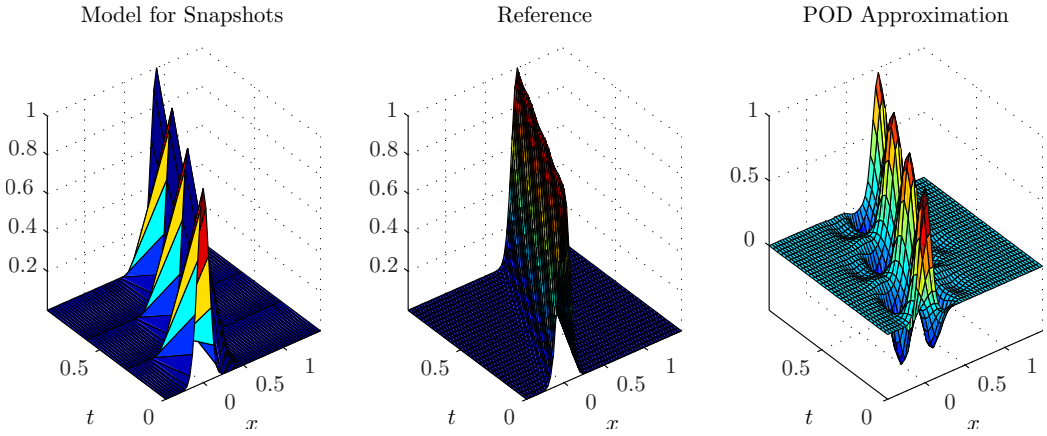

Figure 6.10: Interpolation of the "challenging example" for $a = 100$. The quality of approximation becomes quite poor. Snapshots are taken on a coarse grid. The low-order solution ("POD Approximation") is computed on the (fine) grid of the reference solution.

Note that for these choices it holds $Z(x_0, t{=}0) \approx 10^{-9}$. Hence, we may assume *homogeneous* Dirichlet boundary conditions as considered throughout the thesis. The snapshot model, the reference solution as well as the POD approximation are depicted in Figure 6.9.

Investigation for Fast Decrease Let us now choose the parameter considerably higher. According to the experiments in Subsection 6.1.3, this shall present problems to the POD Method. This fact may be seen in the result in Figure 6.10. In particular we have chosen

$$Z_a(x,t) = \exp(-a(x-t)^2), \quad a = 100, \quad x \in [-0.5 : 0.033 : 1.5],$$

where again homogeneous Dirichlet boundary conditions are applicable since $Z(x_0, t{=}0) \approx 10^{-11}$.

Dependence on Snapshot Location We wish to illustrate that the error in the representation depends on the relative position to the snapshots. In the same breath, we show the dependence of the error on the choice of the parameter a. In particular, we compare the solution at time instances, where snapshots were taken, with time instances between those "snapshot time instances". The

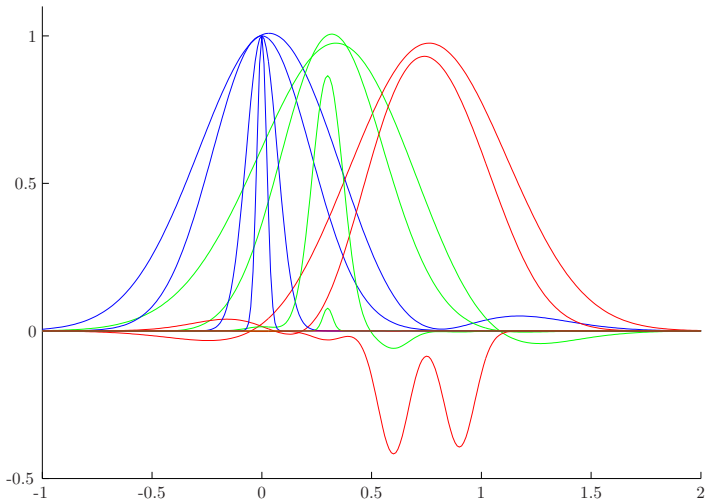

Figure 6.11: POD Solution for $a \in \{5, 10, 100, 1000\}$ at time instances $t = 0$ (blue), $t = 0.33$ (green) and $t = 0.75$ (red), i.e., at the snapshot time $t = 0$, close to the snapshot time $t = 0.3$ and *in between* of the snapshot times $t = 0.6$ and $t = 0.9$.

respective problem data read

$$Z_a(x,t) = \exp(-a(x-t)^2), \quad a \in [5, 10, 100, 1000], \quad x \in [-2 : 0.01 : 3].$$

In Figure 6.11, we see that the quality of the approximation depends on the relative position to the snapshots. (The respective reference solution would be a shift of the "blue plot" to the "right" by the respective value for t.)

Conclusions The higher a, the more the POD Method struggles to capture essential structures of this "challenging example". Consequently, the quality of interpolation decreases with increasing a. It is best close to the snapshot locations.
In contrast, we would obtain "perfect" results for problems whose solutions are made up of (say) a few Fourier modes in space. In this case, the spatial properties would be captured "exactly" and the temporal evolution would be as exact as the (temporal) solution of the reduced-order model, i.e., as exact as the reference solution.
Summarizing, we may say that the method presents a way of interpolating an existing solution by means of its *characteristics* (in contrast to "general" polynomial interpolation say). The *quality* of this interpolation depends on whether the characteristics of the "solution to be interpolated" are already present in the information on the solution provided.

6.2.4 Applying POD with "Perturbed" Snapshots

The main question in this context is: When is a problem *"similar"* enough to another problem such that a POD Approximation to the latter leads to a good approximation to the former?
We shall investigate basic types of perturbations and in this way also point out a "characteristic" of the POD Method. Theory-wise, this property is implied by the fact that the POD Method is based on the "vertical approach" for IVPs, i.e., that the POD Method is to capture "key *spatial* structures".

Procedure Generally speaking, we focus on the illustration and the "idea" – and do not carry out a thorough error analysis say. We shall use simple examples to (again) stress the "nature" of POD-ROM. In particular, we illustrate that the POD establishes a basis of *spatially* important ingredients and the resulting ROM determines the *temporal* coefficients. I.e., we show that it is *crucial* whether

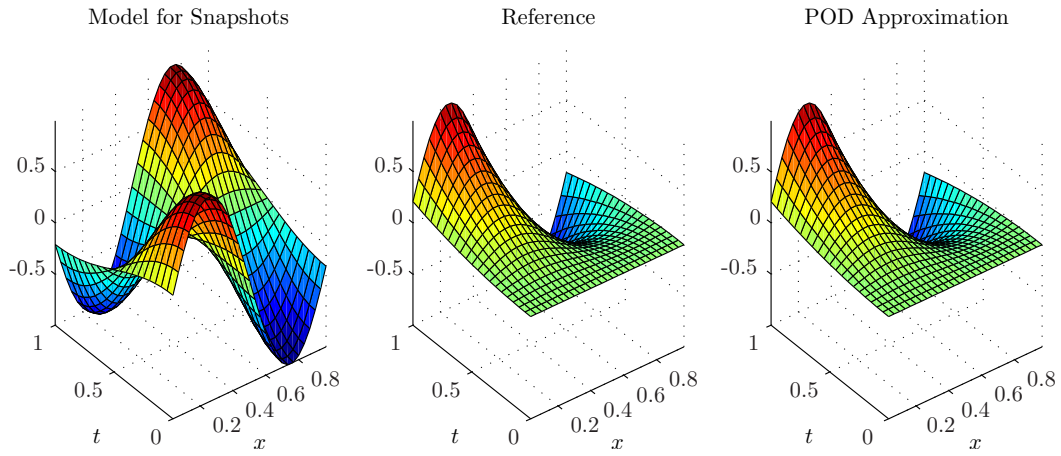

Figure 6.12: Snapshot set Z^1_{snap}: The *spatial* modes in the snapshot set and the reference solution coincide; the time evolution is different. The POD low-order solution is in very good agreement with the reference solution.

the perturbation takes place in time or in space. Furthermore, we shortly investigate a perturbation in the initial value.
Throughout this subsection, we as reference solution and discretization use

$$Z(x,t) \;=\; \sin(2\pi x)\, t^2, \quad y_0 = 0, \quad x \in [0 : 0.033 : 1], \quad t \in [0 : 0.05 : 1].$$

Modes vs Coefficients We consider two snapshot sets, generated by

$$Z^1_{\text{snap}} \;=\; \sin(2\pi x)\, \cos(\pi t) \quad \text{and} \quad Z^2_{\text{snap}} \;=\; \sin(3\pi x)\, t^2.$$

Note that Z^1_{snap} generates a solution of the same modes, i.e., the time evolution is perturbed. This vice versa holds for Z^2_{snap}.
As the POD aims to find suitable *spatial* ingredients for the ROM and the time evolution is determined by the low-order model, we find that the POD Method works fine for Z^1_{snap} and *fails* for Z^2_{snap} (see Figures 6.12 and 6.13, respectively).
In the Z^2_{snap}-case, the POD Solution is "numerically" zero, which basically reflects the fact that the modes in Z^2_{snap} and the reference solution are orthogonal: The projection of the reference system on the POD Basis (i.e., the "key features" of Z^2_{snap}) is zero.

Perturbation in Initial Value We still consider the reference solution above. In this case, we actually take our snapshots from a numerical simulation of an IVP of the heat equation. We endow this problem with the same RHS as the reference problem, but change its initial value to

$$y_0 = \sin(3\pi x) \quad \text{instead of} \quad y_0 = \sin(2\pi x).$$

Hence, a reference solution is not that easy to find (which is basically why we chose to obtain the snapshots by a numerical simulation). In Figure 6.14, we see that the POD approximation leads to a good result.
This is not exactly surprising since the different initial value simply yields another component in the snapshot set – which is orthogonal to all others present. Hence, in the projection process it simply is "ignored".
Furthermore, it is a typical property of parabolic problems that perturbations in the initial value are not carried on in time (refer to Proposition 4.1.5). It would hence be desirable to (in contrast) investigate the behaviour of hyperbolic problems, for instance.

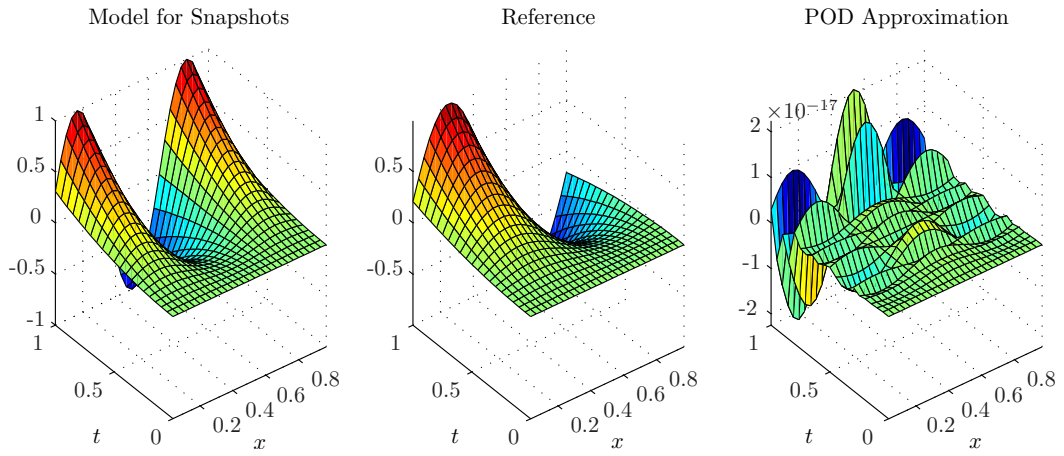

Figure 6.13: Snapshot set Z^2_{snap}: The *spatial* modes in the snapshot set and the reference solution are orthogonal. Even though the time evolutions coincide, the POD approximation is "numerically" zero (note the scale of 10^{-17} of the z-axis).

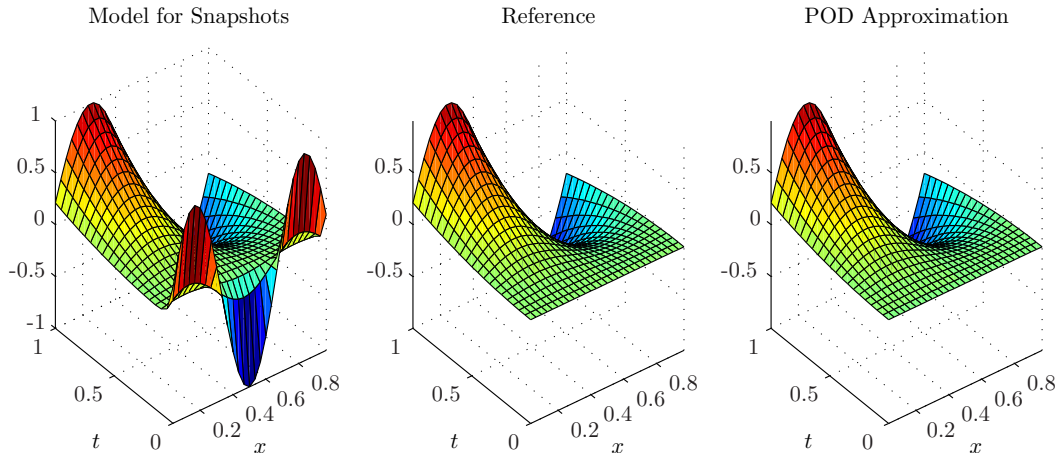

Figure 6.14: The POD Method applied to a snapshot set with perturbed initial value.

Conclusions By means of two "extreme" examples, we have found that the quality of the POD approximation critically depends on the "type" of perturbation. The method might be very sensitive to perturbations in the *spatial* characteristics but quite robust to perturbations in *time* (for the reasons given above).
A perturbation in the initial value may simply introduce an additional "component" into the snapshot set and for the case of parabolic problems does not present a problem (see above).

6.2.5 Study: Influence of the Error in the Snapshots on the POD Basis

In the error analysis, we have assumed to *know* the exact solution and hence assumed to have "ideal" snapshots. In this subsection, we wish to investigate the influence of the errors in the calculation of the snapshots on the resulting POD Basis.

Procedure We choose the data in an IVP for the Heat Equation such that the following function becomes its reference solution:

$$Z_a(x,t) = \exp(-a(x-t)^2), \quad a \in \mathbb{R}^+, \quad x \in [-2 : 0.2 : 2], \quad t \in [0 : 0.2 : 1].$$

(This coincides with the "challenging" example from above. But note that we have chosen the spatial domain such that we can assume that the problem admits homogeneous Dirichlet boundary conditions.) We take snapshots from this reference (just as we did in the theory) as well as from a numerical solution of the IVP. For both the snapshot sets, we carry out POD to capture 99% of the energy. We then wish to find out how well the "numerical" POD Basis matches the "analytical" one.

Matching Two Basis It is somehow "tricky" to compare two basis. Actually, we are interested in how well the *linear spans* of the basis match each other. It shall suffice however to measure how well the analytical basis may be represented by the numerical one. We measure this ability in a *"linear regression problem"*.
Let $\mathcal{B} := \{\psi_i\}_{i=1}^{\ell}$ be the analytical POD Basis and $\mathcal{B}_n := \{\varphi_i\}_{i=1}^{\ell_n}$ the numerical one. Then, the estimation problem reads

$$\min_{x \in \mathbb{R}^{\ell_n}} \left\| \left[\varphi_1, \varphi_2, \ldots, \varphi_{\ell_n}\right] x - \psi_i \right\|_2^2, \qquad \text{for } i = 1, \ldots, \ell.$$

Carrying out a well-known QR-decomposition on the respective matrix, we may easily find the minimal value for each $\psi \in \mathcal{B}$. (The "solution" x would then denote the coefficients of the basis elements of $\mathcal{B}_n$ that would yield an optimal representation of $\mathcal{B}$.)

Discussion of Errors The minimal value found by the QR decomposition represents the error in the representation of one element of $\mathcal{B}$ by the elements of $\mathcal{B}_n$. We may then calculate a *weighted mean* of these "errors" for all elements in $\mathcal{B}$. As weights, we use the corresponding energy contributions since we wish to especially consider errors in the "important" directions. Let us refer to this "average of errors" as the *"error of representation"*.
Note that the error of representation actually is composed of two errors: the error due to calculating the snapshots as well as the error due to the calculation of the POD Basis. Surely, we actually are interested in the error introduced by the POD Method. Let us therefore also investigate the *contribution* of the actual POD Basis error to the error of representation. (In other words, we wish to examine which proportion of the error of representation is actually due to the (numerical) errors in the snapshots.)

Results We set $a := 100$ and use:

$$h \in [0.02 : 0.005 : 0.2] \quad \text{and} \quad \tau \in [0.01 : 0.0025 : 0.1].$$

In Figure 6.15, we see the resulting error of representation, depending on h and τ. We find that for sufficiently small mesh sizes h and time step sizes τ, the relative error (weighted by energy) is of the magnitude of 1% – which is quite satisfying.
On the other hand, for small values of h, the "POD Basis error" amounts a larger proportion of the "error of representation" than for large values of h (see Figure 6.16).
Altogether, it turns out that for the example chosen, the POD Method is quite robust – even though the example had been a "challenge" to the POD Method in other circumstances (see above).

6.3 Numerical Approaches to Suboptimal Control

Let us in this section use the ROM in order to carry out suboptimal control approaches for the non-stationary heat equation.
We restrict ourselves to "distributed control" and do not impose constraints on the control. (This would require a careful consideration of algorithms of constrained optimization – which is not the focus of this thesis.)

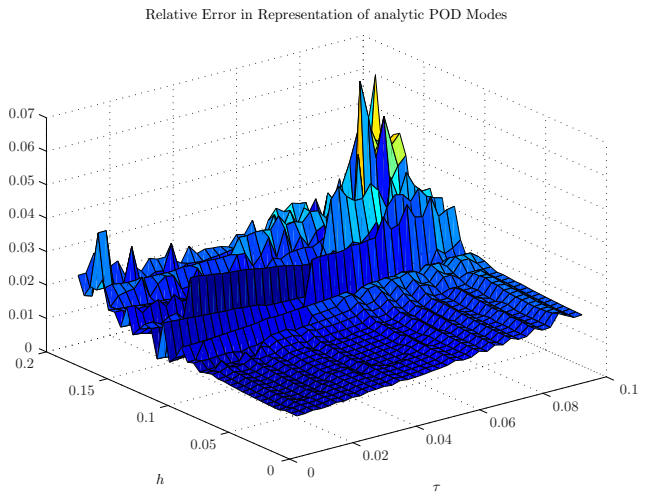

Figure 6.15: Relative, energy-weighted error of representing the "analytic" POD Basis in terms of the "numerical" POD Basis.

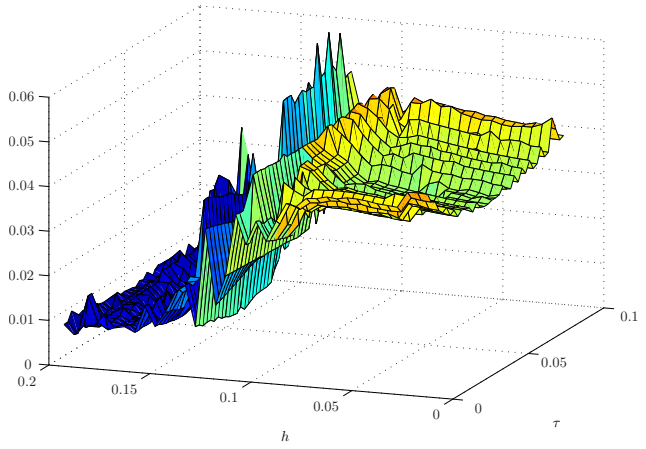

Figure 6.16: Relative contribution of the "POD Basis error" to the error of representation.

Procedure We first quickly denote the actual problem of concern. Then, we investigate an example for a final value control which we extend to a "full state tracking" problem. In this context, we again consider the "challenging example" of the previous sections. Throughout, we use an FE scheme to obtain a "reference" solution.

6.3.1 Optimal Control of the Heat Equation

The non-stationary heat equation is a special case of Problem 5.2.6. Thus, we may obtain the optimal control problems as well as its treatment from Subsection 5.2.4.

Problem Statement For a *distributed* control $u \in \mathcal{U} := L^2(Q)$, a "full state target" $y_Q \in L^2(Q)$ as well as a "final value target" $y_\Omega \in L^2(\Omega)$ we obtain the problem

$$\min_{u \in \mathcal{U}_{\text{ad}}} f(u) := \frac{\alpha_1}{2} \int_0^T \|y(t) - y_Q(t)\|^2_{L^2(\Omega)} + \frac{\alpha_2}{2} \|y(T) - y_\Omega\|^2_{L^2(\Omega)} + \frac{\sigma}{2} \|u\|_{L^2(\Sigma)},$$

where the pair (u, y) fulfills an IVP for the non-stationary heat equation.

$$\begin{aligned} \frac{\partial}{\partial t} y - \Delta y &= u \quad \text{in } Q, \\ y &= 0 \quad \text{in } \Sigma, \\ y(0) &= 0 \quad \text{in } \Omega. \end{aligned}$$

The constants α_1, α_2 and σ shall be chosen later on. Note that for very small σ, the numerical solution may become instable.

Optimality Conditions Since we do not impose constraints on the control, we immediately infer from Corollary 5.2.7 (in particular from (5.10)) that we may obtain the control from

$$u = -\frac{1}{\sigma} p,$$

where p denotes the *adjoint state*, i.e., solves the IVP

$$\begin{aligned} -\frac{\partial}{\partial t} p - \Delta p &= 0 \quad \text{in } Q, \\ p &= 0 \quad \text{in } \Sigma, \\ p(T) &= y(T) - y_\Omega \quad \text{in } \Omega. \end{aligned}$$

By inserting the expression for u into the state equation, we obtain a *coupled system of IVPs* for the state as well as the adjoint state. We may then solve the resulting linear system according to Subsection 5.3.3.

Choice of Snapshot Set – Simplification We only wish to *demonstrate* that the POD Method actually may calculate optimal controls faster than an FE approach. Therefore, we choose a very simple environment to apply the POD Method in. In particular, we compose our snapshot set of the *requirements* on the state in the control problem. This is to say that in the case of "final value control", the snapshot set consists of the initial value as well the desired final value. In case of "full state control", we let the snapshot set consist of snapshots of the desired state.
Furthermore, we choose the POD Basis computed for the state to be used for the control as well.

Discussion of Simplification In our setting, there is no need to take snapshots of the state. Hence, when considering the numerical effort, the time for *obtaining* snapshots as well as possible *errors* in them would have to be taken into account additionally.
If we assume that the desired state is "reachable", i.e., that there exists an optimal control such that the optimal state coincides with the target state, our snapshot set contains characteristics of the optimal state (since it is made up of snapshots of the target state). Therefore, the problem of non-modeled dynamics is overcome and there is no need to apply the adaptive POD algorithm.
Using the POD Basis of the state for the control as well is only applicable if the control admits similar characteristics as the state does. In our example, this would be true for "eigenfunctions" of the differential operator for instance. In view of the discussion of POD suboptimal control, this procedure may be interpreted as setting up only *one* POD Basis for state and adjoint state but taking snapshots of the *state only*. (Recall that the control in this example is *proportional* to the adjoint state.)
The previous problems notwithstanding, we find satisfying suboptimal controls in our experiments (see below).

Objectives of the Experiments In the following experiments we shall illustrate the following theoretical findings:

1. In order to minimize the computation time of the basis elements, it is important to choose either the Method of Snapshots or the Classical Approach according to the "relative magnitude" of the numerical discretization and the number of snapshots.

2. The FE appraoch is "symmetric" in time and space, i.e., both the discretizations determine the size of the optimality optimality system equally.

3. For POD optimal control the size of the optimality system depends on the number of modes used (in the examples, always two or three) as well as the number of time steps in time discretization.

4. For this academic example, the quality of the POD suboptimal control depends mainly on the ability of the POD Method to represent the target state.

Investigation of the Advantage of POD Control The size of the FE optimality system is determined by the time as well as the space discretizations equally. On the other hand, the size of a POD optimality system depends on the number of modes used as well as the number of time steps in the low-order solution but is independent of the degrees of freedom in the spatial discretization.
Additionally, the size of the eigenvalue problem (of the Method of Snapshots) to find a POD Basis is determined by the number of snapshots. Therefore, the basis computation time is not increased significantly by increasing the spatial degrees of freedom.
Hence, the POD Method (in particular the Method of Snapshots) is especially of advantage for fine discretizations in space.

6.3.2 Final Value Control

Essentially, we try to reconstruct the state of the "challenging example" from its initial- as well as its final value.

"Snapshot" ROM – POD Not Necessary We let our snapshot set consist of only two members (which shall become the initial value as well as the target in our control problem):

$$y_0(x) := 4x(1-x) \quad \text{and} \quad y_\Omega(x) := \sin(2\pi x).$$

Since the two modes do not have much in common, two POD modes are extracted in order to establish a POD Basis. Due to the choice of weight, both the modes approximately contribute 50% of the total energy. As a consequence the POD Method might simply be omitted and the "snapshots" itself might be used as a basis for the ROM.

Actual Choices in the Problem Since we wish to track the final value y_Ω, we choose $\alpha_1 = 0$ and $\alpha_2 = 1$. Furthermore, we set $\sigma = 10^{-6}$ and (for various choices of h and τ) discretize the IVPs according to

$$x \in [0 : h : 1], \quad t \in [0 : \tau : 1].$$

Quality of the Approximation As we can see in Figure 6.17, for $h = \tau = 0.05$ the full-order control and the POD control are in very good agreement. In particular, the relative error of the POD final state to the FE final state amounts to 1.62×10^{-7}.

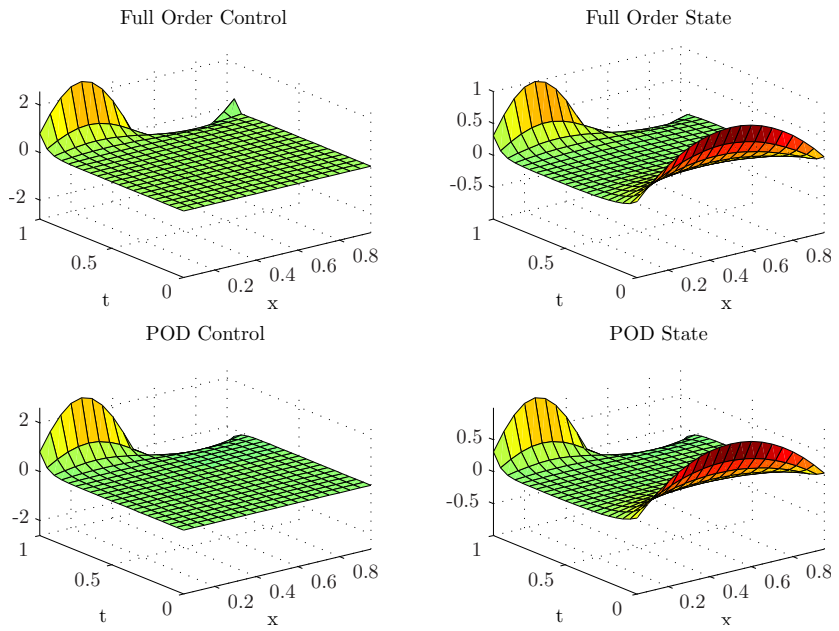

Figure 6.17: Full-order and POD "final value control" with $h = \tau = 0.05$.

Investigation of the Benefit of the POD Method Let us now study whether the application of the POD reduces the numerical effort to obtain the "same" solution and hence reduces the computation time for finding an optimal control.
Recall that there have been three ways to compute a POD solution, where in the FE case we have proposed to use the Method of Snapshots (refer to Theorem 3.1.5 and Corollary 3.2.1, respectively). Note that this choice was based on the general assumption of having more degrees of freedom in space than snapshots (in time). – In this "artificial" test however, we shall consider three cases and will also consider a case in which the Method of Snapshots actually is *not* suitable. In particular, we find in this case that the POD Method does *not* even decrease the time to compute an optimal control. In contrast, the method is *very* effective in the other two cases.
In particular, we find for the three cases (also see the corresponding "data" in Table 6.3):

1. "Coarse Space Discretization/Many Snapshots": The time to compute the POD Basis is so long that the FE approach is way quicker.

2. "Balanced Amount": The POD Method takes about 10% of the time needed by the FE scheme. (The absolute amount of time measured is short, which might make them less reliable.)

3. "Fine Space Discretization/Few Snapshots": This is the most common scenario the POD Method would be applied in. The reduction in computation time is about 98% compared to the FE approach.

In fact, the POD Method always manages to extract two basis functions, capturing 99% of the energy – which is *not* an actual reduction of snapshots (see above). Hence, in all cases, the optimality system is way smaller than the one in the FE approach. Yet the computation of the basis functions is time consuming itself and therefore, the FE approach might actually be quicker (refer to the first case).
To be fair, we should mention that the first case is unrealistic in numerical simulations. For that reason, we use the Method of Snapshots throughout the experiments. If we would have used the Classical Approach to setup the POD Basis, we would in the first case have obtained similar results as the Method of Snapshots did in the last case.
On the other hand, in a less academic situation, we additionally would have to take snapshots of the solution, consuming time additionally.

Discretization	$h = 0.1$, $\tau = 0.001$		$h = \tau = 0.01$		$h = 0.001$, $\tau = 0.1$	
Method	FE	POD	FE	POD	FE	POD
Size of System	18018	4004	19998	404	21978	44
Solution Time (s)	0.4530	0.1570	1.2030	0.0160	*2.3590*	≈0
POD Basis (s)	0	*72.5470*	0	0.0930	0	0.0470

Table 6.3: Comparison of the numerical effort of computing a final value control by a Finite Element (FE) as well as a POD system ($\ell = 2$) for various choices of discretization. The POD Basis was computed via the Method of Snapshots.

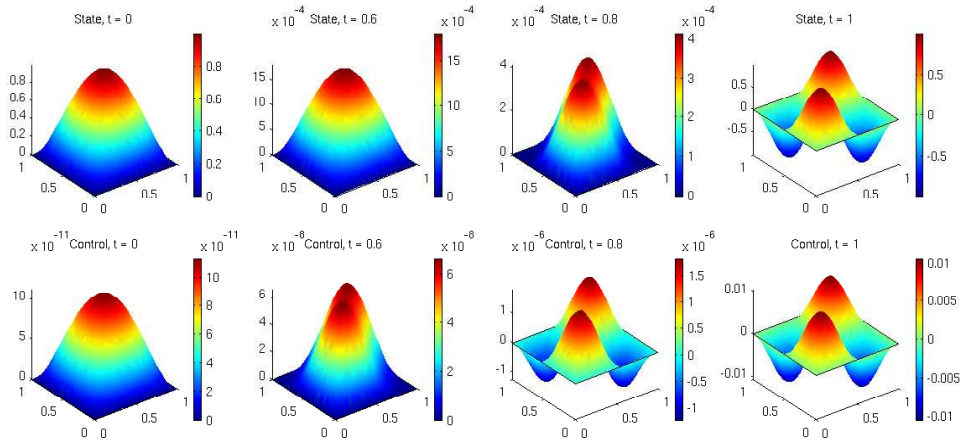

Figure 6.18: Visualization of the POD final value control in two space dimensions. Depicted are the state (upper row) as well as the control (lower row) at the time instances $t = 0$, $t = 0.6$, $t = 0.8$ and $t = 1$.

Summing up, we find that for this example, POD *does* reduce the computation time significantly (if applied correctly).

Example in two Space Dimensions The setting coincides with the one dimensional example above. We only choose $x \in [0,1]^2$, triangulated with a maximum mesh size of $h = 0.05$ as well as two dimensional functions for the initial value as well as the target:

$$y_0(x,y) := 16x(1-x)y(1-y) \quad \text{and} \quad y_T(x,y) := \sin(2\pi x)\sin(2\pi y).$$

In the FE case, we obtain $q = 885$ degrees of freedom which yield an optimality system of 19470 variables. The solution time totals 11.9 seconds. The POD Method (due to the construction of the problem) captures 99% of the energy by two modes. The basis is computed within 0.7 seconds and the optimality system, involving 44 variables, is solved within 0.05 seconds. Therefore, the POD solution takes only about 7% of the time needed for the finite element solution.

We have depicted the respective states and controls at four time instances in Figure 6.18 for the POD approach and in Figure 6.19 for the FE approach. In both cases, the final value reached is in good agreement with the target.

Note however that the FE control significantly differs from the POD control. This might be due to the fact that we have simplified the problem *too much* by choosing the basis for the possible controls to consist of the POD modes. In particular, the figure of the FE control for $t = 1$ "looks" like this FE control does not lie in the span of the POD Basis at this time instance.

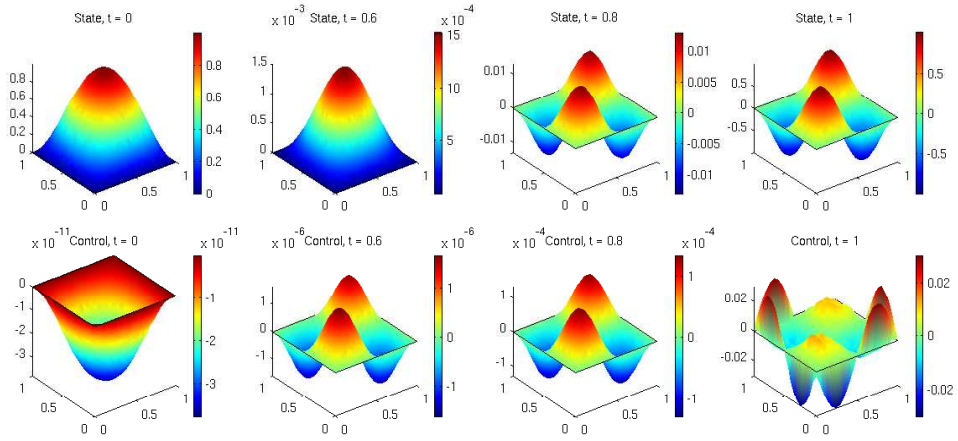

Figure 6.19: Visualization of the FE final value control in two space dimensions. Depicted are the state (upper row) as well as the control (lower row) at the time instances $t = 0$, $t = 0.6$, $t = 0.8$ and $t = 1$.

6.3.3 Full State Control

In this subsection, we wish to track a full state over time. We choose two types of targets, one for which we expect the POD Method to perform well as well as a more challenging one.
Note that since "all the dynamics" in the system may be controlled, we may construct the suboptimal control approach from a POD Approximation of the target state. I.e., there neither is need to take snapshots from any system nor to think about a basis update (see discussion above).
The quality of the suboptimal control hence is directly linked to the ability of the POD Method to approximate the target state.

Easy Example – Fourier Modes We let our target state be made up of Fourier modes. According to Subsection 6.1.2, we then expect the POD Method to represent the state rather well, i.e., to lead to good results for the suboptimal control. In particular, we choose

$$Z(x,t) = 4x(1-x)(1-t) + \sin(2\pi x)t + \sin(4\pi x)\sin(2\pi t), \quad x \in [0 : 0.033 : 1], \quad t \in [0 : 0.033 : 1].$$

Note that this choice presents an "extension" of the previous initial-/final- value to the whole time domain. We find that this works out particularly fine (as expected) – refer to Figure 6.20. Furthermore, we choose the weights in the objective to be $\alpha_1 = \alpha_2 = 1$ and $\sigma = 10^{-6}$.
As mentioned above, we may use the target state to generate snapshots. For the sake of simplicity, we obtain snapshots of this state on the same time grid as for the solution of the optimal control problem. Therefore, the parameter τ (implicitly) also determines the number of snapshots taken.

Results for Fourier Target State The POD Method extracts three modes out of the 31 snapshots (time instances of the target state). Hence, in this case the POD Method is *useful* to apply. Based on these snapshots, the suboptimal control is calculated well – as we may see in Figure 6.20.
Of course, we could again investigate the numerical effort for different choices of τ and h. Yet the full order optimality system is not changed in size and the POD optimality system is increased slightly due to the increased number of basis functions used. Hence, we would obtain very similar results in this context. Note however that we are given more snapshots and hence the computation time of the POD Basis may increase even further: For $\tau = 0.001$ and $h = 0.1$, the basis computation time rises to about 107 seconds, but could be decreased by using the Classiclal POD approach (see "1D final value" example).

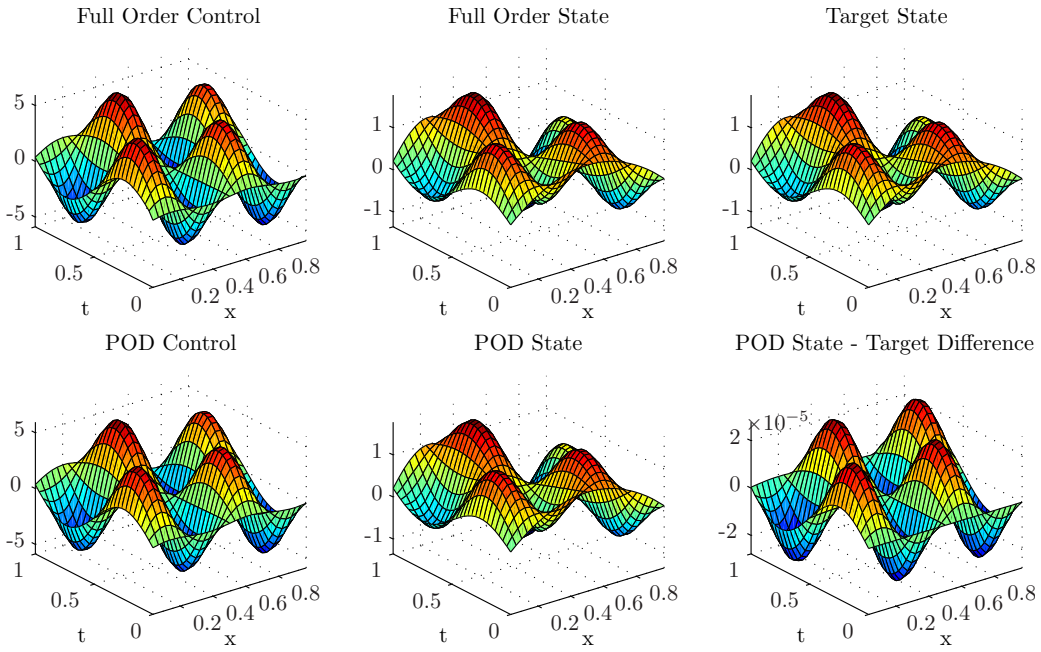

Figure 6.20: POD vs full-order control targeting a state consisting of Fourier modes. The POD state is in very good agreement with the target (note the scale of 10^{-5} of the z-axis of the lower right figure).

Challenging Example We make use of the "challenging" example used throughout the experiments:

$$Z_a(x,t) = \exp(-a(x-t)^2), \quad a := 100, \quad x \in [-0.5 : 0.033 : 1.5], \quad t \in [0 : 0.033 : 1].$$

The POD Method captures 99% of the energy by means of 10 modes. Yet as depicted in Figure 6.21, the quality of our suboptimal solution is not as good as in the previous example. (There are small oscillations).
This was expected since we have learned in Subsection 6.1.3 that the POD has got difficulties in representing the particular target state (in contrast to the Fourier state used above). The actual error in the target state representation however is surprisingly good.
Again, we furthermore wish to investigate whether the POD Method actually increases the speed of calculation. For $h = \tau = 0.01$ (i.e., 100 snapshots), we find that the POD approach only needs 3.4% of the FE computation time, which for $h = 0.001$ and $\tau = 0.1$ reduces to about 3%. Details may be found in Table 6.4.
Hence, even for this more challenging example, the POD Method reduces the computation time significantly. (Let us again stress however, that we did not have to compute snapshots beforehand.)

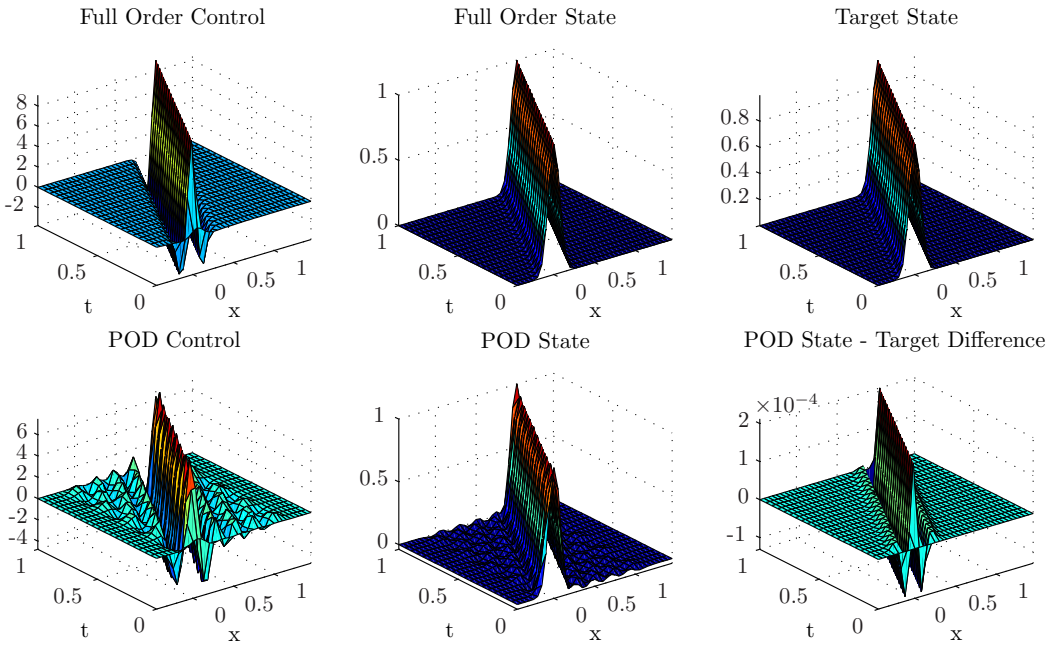

Figure 6.21: The POD has got difficulties in representing the target state of the "challenging" example.

Discretization	$h = \tau = 0.01$		$h = 0.001$, $\tau = 0.1$	
Method	FE	POD	FE	POD
Size of System	40198	2020	43978	220
Solution Time (s)	*2.969*	0.078	*8.281*	0.031
POD Basis (s)	0	0.234	0	0.203

Table 6.4: Comparison of the numerical effort of computing a "challenging" full state control by a Finite Element (FE) as well as a POD system ($\ell = 10$) for two choices of discretization.

Chapter 7

Summary, Discussion and Outlook

In this chapter, we shall summarize all our findings regarding the objectives proposed in the introduction. Then, we shall discuss the chances of the POD in Model Reduction as well as in Suboptimal Control. Extracting the major problems, we comment on possible improvements of the POD Method.

7.1 Summary

Let us summarize our findings according to the objectives of the thesis introduced in the introduction and depicted as gray boxes in Figure 1.
The (extended) discussion shall be touched upon in the next section. For all other objectives, there is a corresponding subsection (in this section) and each paragraph corresponds to one aspect of the objective (i.e., to a node in Figure 1).

7.1.1 Results on Optimal Control

We were mainly concerned about introducing the problem and finding (numerical) ways of solution for a particular case.

Optimal Control We formulated a linear-quadratic control problem for Evolution Problems mathematically and found *optimality conditions* on a continuous level. These consist of a so-called "adjoint problem" and an "optimality condition". In the control non-constrained case, the system reduces to a *coupled* system of state and adjoint equation.
In terms of numerical treatment, we transformed the Optimal Control problem into a standard quadratic programming problem ("discretize then optimize"). Alternatively, we discretized the optimality system derived and explained a "gradient projection" method to solve it ("optimize then discretize"). We pointed out that (say) a Newton-type method would be better suited to the problem. Furthermore, we denoted the discrete coupled system corresponding to the case of no control constraints.

Feedback We focused on the "linear quadratic regulator" problem and showed that the solution is given by an equation of matrix Ricatti type. Furthermore, we drew a theoretical link to open loop control.

7.1.2 Findings for the POD Method

We investigated the theory of the POD Method in two parts: In the first part, we presented the general theory to be used in context of reduced-order modeling whereas in the second part, we

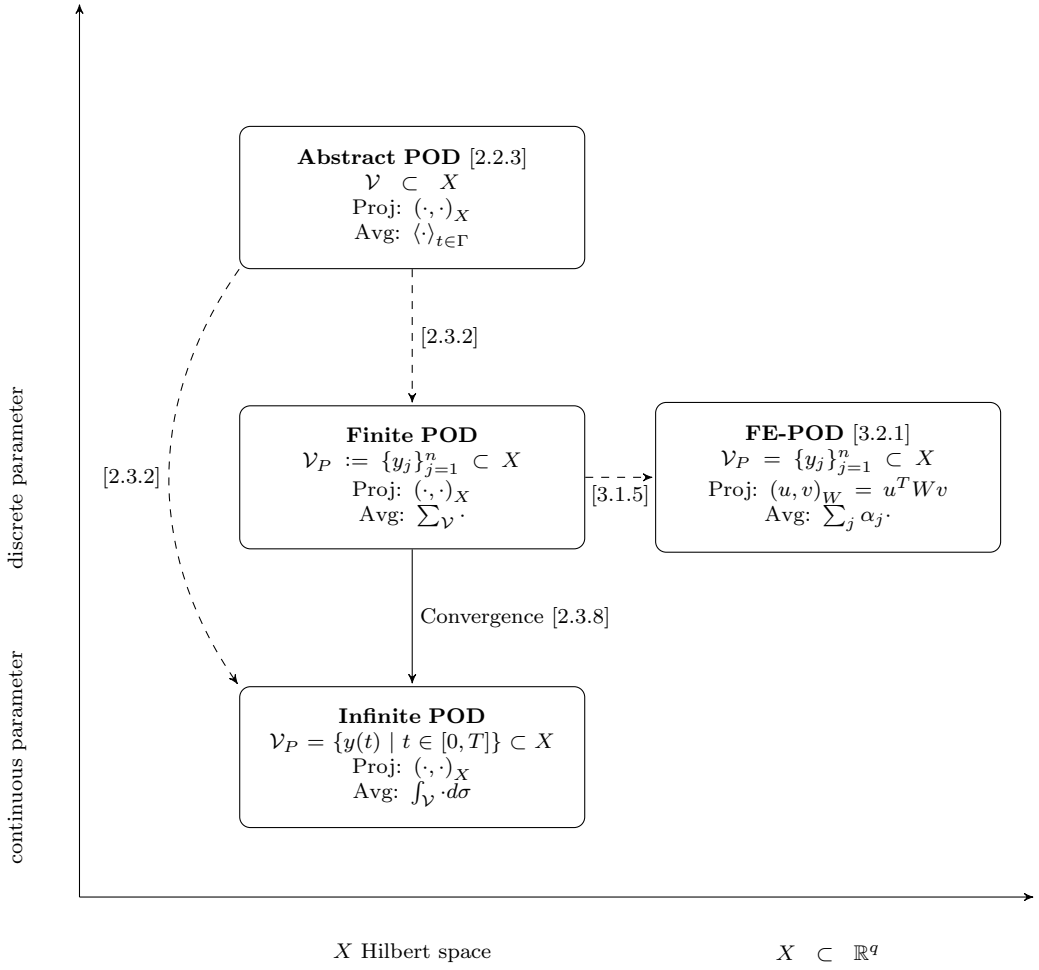

Figure 7.1: Overview of all POD Problems treated. In each note the name of the problem, the ensemble set, the projection operator and the average operator are denoted. Dashed arrows depict deductions of solutions of problems. The numbers in brackets indicate the theorem, proposition or corollary in which a respective solution has been established.

presented further insides on the POD Method, not necessarily closely linked to the remainder of the thesis.

Theory on the POD Method We constructed a POD Problem according to the idea that the resulting modes should "optimally" represent a given "snapshot set". In particular, we defined a POD Basis to be an orthonormal basis of a subspace $\mathcal{B}$ such that the best approximation of a snapshot in $\mathcal{B}$ should be minimal on the average over all snapshots. The best approximation is measured in an "optimality norm" which remains to be chosen.

On an abstract level, we have set up a *characterization* of a POD Basis as a subset of eigenvectors of the "POD operator". This reduced the existence of a POD Basis to the (Hilbert Schmidt) theory of compact linear operators in a separable Hilbert space.

In the remainder we deduced POD Problem statements in several contexts. We have depicted a respective overview in Figure 7.1.

Further Insides by means of Statistics For ensembles of "abstract functions", we found that the POD modes are actually parts of a "bi-orthogonal decomposition" of the ensemble. We hereby justified the "Method of Snapshots", an alternative way to calculate a POD Basis. By means of this, we slightly have brought together two "historically" different approaches to the POD Method. In particular, the role of the statistical approach in the numerical one was outlined.
We introduced the field of dynamical systems, where the focus generally is "wider": The *fundamental* structure of evolution is of concern. We interpreted the POD operator as a "correlation" and showed that extracting "coherent structure" presents an *alternative objective* of the POD Method.

7.1.3 Review of POD Suboptimal Control and Numerical Results

Let us review the main steps in a Suboptimal Control strategy. In particular, we wish to draw links to the respective results of the numerical investigations.

POD for Evolution System – Procedure and Findings We explained that, in context of Model Reduction, a suitable ensemble to apply the POD Method to is a *set of snapshots*. We found that the snapshots may be obtained by *observation* of a physical system or may also be determined by a *numerical simulation* of an Evolution Problem. In the latter case, the snapshot set consists of (say) FE vectors of the solution at chosen time instances. For that reason, we choose the inner product in the POD Problem to be a weighted inner product (according to the FE space) and the average operation to be weighted by *trapezoidal weights* in order to ensure convergence in time.
We obtained POD modes in three ways (refer Figure 3.1): SVD, "Classical approach" and "Method of Snapshots", where the latter method is the method of choice for FE snapshots. Furthermore, we mentioned improvements of the snapshot set (such as the reduction of the mean from the snapshots). (A summary of applying the POD Method in context of FE snapshots may be found at the end of Subsection 3.2.3.)
In numerical experiments, we found that in case the snapshot set consists of Fourier modes, a POD representation works out fine for a POD Basis of quite a low rank. In this context, we also illustrated the notion of "energy": energetically low contributions are "ignored" by the POD Method. For a "challenging example" (of quickly traveling information), the quality of representation was acceptable as well since the rank ℓ is chosen according to the amount of energy which is to be captured. In comparison to the "Fourier example", ℓ was increased, i.e., the *order* of a satisfying approximation was not as low as in the previous case.
In a study, we found that *subtracting the mean* from a snapshot set may reduce the order of approximation by one. Additionally, this may avoid "difficulties in scaling" in case that the fluctuations in the snapshot set are energetically small in comparison to the mean of the snapshots.

Reduced-Order Modeling We chose to treat Evolution Problem "vertically", with a Galerkin ansatz in the space discretization. We provided a low-dimensional (spatial) ansatz composed of *key spatial ingredients*. This ansatz in comparison to general ansatz spaces yields a *low-order model*. As outlined, we determine those "key spatial ingredients" by the POD Method applied to (temporal) snapshots of the Evolution Problem.
In this sense, the POD Method presents a "discretization concept" and the corresponding low-order model is to determine the coefficients of the solution w.r.t. the POD Basis.
Finally, we investigated *practical scenarios* in which POD-ROM may be of help:

1. The method presents a way of *interpolating* an existing solution by its characteristics (in contrast to (say) "general" polynomial interpolation). As expected, the quality of approximation is best close to snapshot positions. It furthermore depends on whether the characteristics of the "solution to be interpolated" are already present in the solution provided (which we showed for the case of *extrapolating* a given solution).

2. We considered the case that the snapshots are known for a system with "perturbed" data. We found that a perturbation in the *initial value* in the worst case introduced an additional "component" into the snapshot set. Due to the "damping" of the initial value in our problem (see (4.9)), the energy contribution to the snapshots may even be so low that the POD Basis is not changed at all.

   Other than that, the POD Method shall extract "key *spatial* structures" whereas the (corresponding) *temporal* structures are to be determined by the resulting reduced-order model. Therefore, the *temporal evolution* of *spatial characteristics* in the snapshots should not be of importance. We justified this assumption for two "extreme" examples in which a temporal perturbation did not influence the result whereas a perturbation in the spatial structure caused the Method to fail.

Apart from that, we studied the error in the POD Basis obtained from snapshots taken of the Heat Equation ("challenging example") in comparison to a POD Basis obtained from a reference solution. For sufficiently small mesh sizes h and time step sizes τ, the relative error (weighted by energy) was of the magnitude of 1%, i.e., satisfying. For small values of h, the "POD Basis error" amounted a larger proportion of the "error of represenation" (in comparison to large values of h).

Suboptimal Control By virtue of the theory on Optimal Control, we directly infer a *sub-optimality system* for "Optimal Control of POD reduced-order models". Since two equations are involved in this optimality system, we considered whether one POD Basis (based on snapshots of both the solutions) would suffice. It turned out that in general it is preferable to maintain two separate POD Basis.
The basic idea of overcoming the problem of "non-modeled dynamics" was to ensure that the POD Basis represented the characteristics of the state which corresponds to the optimal solution (since the cost functional *only* measures these characteristics of the state). We explained that this may be realized by an adaptive control algorithm which regularly *updates* the POD Basis. Alternatively, we may include the above requirement on the POD Basis into the optimality system ("optimality system POD").
We illustrated the Suboptimal Control approach in *simplified* numerical experiments. For a target state consisting of Fourier modes, the results were as satisfying as expected. In the more "challenging" case, we found that the approximation results were surprisingly good as well.
In particular, we found:

1. In order to minimize the computation time of the basis elements, it is important to choose the Method of Snapshots or the Classical Approach according to the "relative magnitude" of spatial discretization and number of snapshots.

2. For the academic example used, the quality of the POD suboptimal control depended mainly on the ability of the POD Method to represent the target state.

3. The (snapshot) POD Method is especially of advantage for fine discretizations in space.

Error Estimation of Reduced-Order Models We assumed that "exact" snapshots were available and that the reduced-order model was discretized in time by the implicit Euler method. Furthermore, the Optimality Norm was chosen to be induced by the bilinear form in the Evolution Problem.
We established two *error estimates* for a POD reduced-order model. In both the estimates, the error consisted of the error due to the time discretization as well as the error caused by "non-modeled energies" in the snapshot set (which we refer to as "POD error").
The asymptotic behaviour of the first estimate with respect to the number of snapshots is not satisfying, whereas in the second estimate the "POD error" is not exclusively estimated by the "non-modeled energies" in the snapshot set. Furthermore, both the estimates depend on the choice of snapshot set. By means of an "asymptotic analysis", we found that the "POD error" estimates are bounded by the estimate of the case of taking the *whole trajectory* as a snapshot set. By virtue of this result and by

including the "difference quotients" of snapshots into the snapshot set as well as further assumptions, we derived a special case of the error estimate which coincides for both the initial estimates. We furthermore refined the context of application of the error estimates (i.a., to FE discretizations).
Since snapshots usually are not known exactly but have to be calculated, we carried out a (spatial) asymptotic analysis in the snapshots. In particular, we found that the "rate of convergence of the spatial approximation in the snapshots" coincides with the "rate of convergence of the corresponding POD operators".

7.2 Concluding POD Discussion

For a final "wrap up", let us gather findings for the aspects of the POD Method which are important in terms of "optimal modeling": its *speed* and its *quality* (of approximation).
Based on these findings as well as other aspects, we wish to discuss the chances of POD in Model Reduction and Suboptimal Control. In the next section, we shall give particular suggestions to the main problems extracted. (Note that the situation is depicted in Figure 7.2.)

What is an "Optimal" Basis for a reduced-order model? Let us define an "ideal objective": An "optimal" basis yields a reduced-order model that provides a solution of desired accuracy. This solution shall be obtained quicker than in any other way. Therefore, we investigate how speed and quality of the POD Method may be influenced.

7.2.1 The Speed of POD

Let us gather all issues of relevance for the speed of calculating a POD reduced-order solution. Roughly speaking, it is determined by the number of snapshots n (size of the eigenvalue problem to solve in order to obtain a POD Basis) as well as the size of the resulting reduced-order model (i.e. the rank ℓ of the basis).

Reduction in Order The POD Method generally is capable of a huge reduction in the order of the discrete approximation. This is due to the "tailored" character of the POD modes and was illustrated on a "matrix level" as well as in numerical calculations.
Furthermore, the POD Method is a linear procedure which lets the basis functions inherit certain properties of the solution (such as divergence freeness). This may also simplify the solution of the reduced-order model (divergence free basis functions in incompressible flow problems, for example).

How to Control the Speed? The rank of the solution may be controlled by the parameter ℓ (which also determines the amount of "energy" of the snapshots to be captured in the POD Basis). A low rank results in a small reduced-order model, i.e., in a short solution time.
Furthermore, the number of snapshots determines the size of the eigenvalue problem, i.e., determines the time to setup a POD Basis. (In numerical examples, we have seen that this time presents a significant contribution to the total time of calculation.)
Usually a major part is spent on the actual *calculation* of snapshots, of course. This issue however cannot be touched upon in this context. Let us only mention that in Holmes, Lumley, and Berkooz 1996, Subsection 3.3.3 it is discussed to *extend* a given snapshot set by means of *symmetries*. This would obviously speed up the process of obtaining snapshots (if applicable).

7.2.2 The Quality of POD

Let us now consider which factors influence the "quality" of a POD approximation (i.e., the "error" in the approximation).

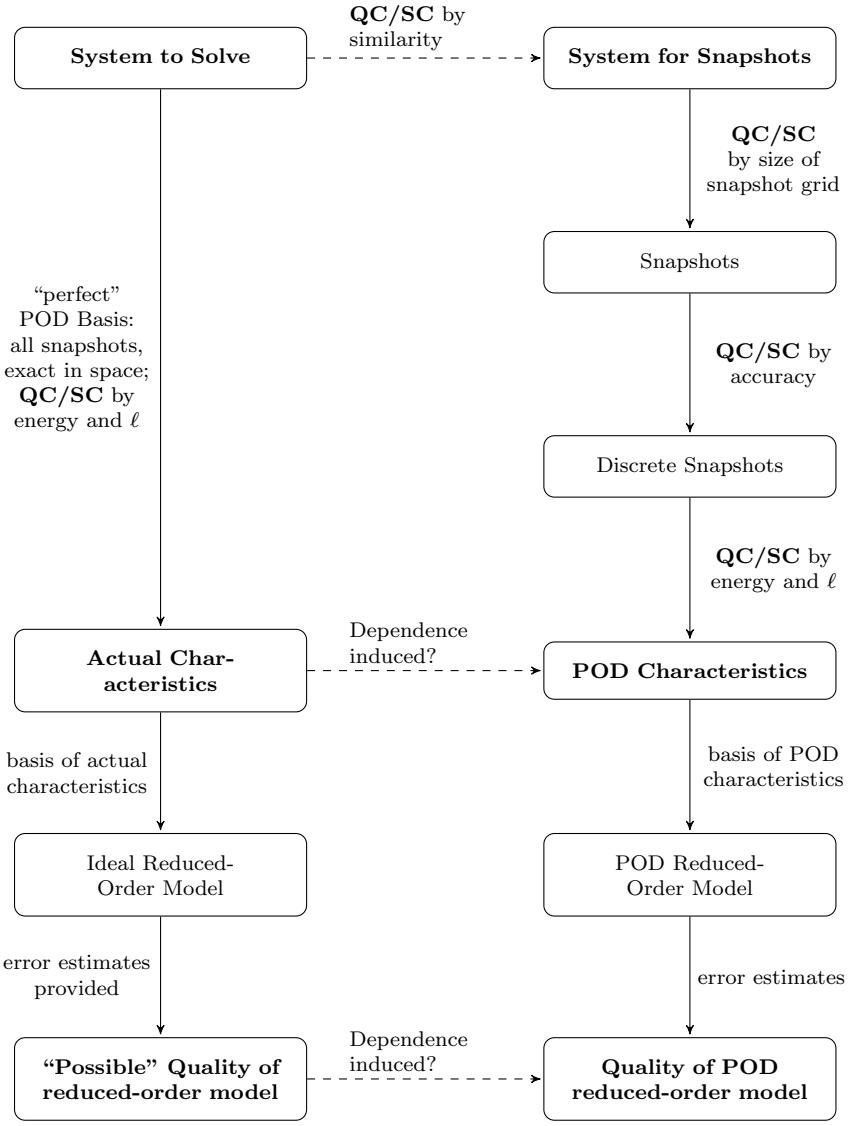

Figure 7.2: Discussion of the Effectiveness of POD. Quality Control (QC) vs Speed Control (SC).

Extracting Characteristics The POD Method extracts *characteristics* of a *particular* snapshot set. In order to obtain the *characteristics* of the actual *solution* of (say) an Evolution Problem, we need to take the whole trajectory of the solution into account – which obviously is not feasible in a practical context.

Capturing Characteristics of the Solution in a Snapshot Set Surely, we wish to capture the *characteristics* of the expected solution in a snapshot set. Only then, the POD Method may be able to "represent the characteristics" of the solution by representing only a snapshot set of it. Unfortunately, we have not found a procedure to determine suitable *locations* of snapshots, i.e., to setup a proper snapshot grid.

In order to obtain snapshots *at all*, some sort of solution to the respective model has to be available. The error in these solutions was not taken into account explicitly.

Capturing Characteristics of the Snapshots in a POD Basis We use the POD Method in order to capture the information in a snapshot set by means of a low rank approximation to them. We have seen that the POD Method provides an "optimal" representation of parametrized data of some (desirably low) rank. In this context, let us stress that the rank of an approximation is *not* to be confused with its information content. The quality of approximation e.g. decreases in case of *"information traveling quickly"* with little "spread in space". A theoretical reason is given by the fact that the "autocorrelation" of the snapshot set ("signal") is very low.
In the numerical examples, we have shown that there actually *are* differences in the capability of the POD Method to represent different snapshot sets: If the snapshot sets consisted of Fourier modes, the method worked out fine. (The POD is a generalization of a Fourier decomposition in some sense.) On the other hand, we considered a "challenging example" on the basis of "quickly traveling information". The POD Method struggled at providing high quality approximations of low rank. In context of parabolic IVP, this case may appear in case of *"dominant convection"*, for instance.

Capturing Characteristics of the Solution in a POD Basis As implicitly outlined, capturing the characteristic information of a solution in a POD Basis is a *two step process*: The information has to be captured in a snapshot set and then this snapshot set is represented by a POD Basis.
Both the processes bear problems which could not be solved in general:

- How to find "characteristic" time instances, i.e., how to find time instances for taking snapshots?
- Do errors in the calculation of the snapshots influence their "characteristics"?
- The POD Method does not work equally well for all types of snapshot sets.

Perturbed Snapshot Sets In numerical examples, we considered the case that the system we obtained the snapshots of and the system we desire to solve do not coincide ("perturbed snapshots"). We found that the POD may be quite robust to perturbations in the snapshots who only affect the temporal structure of the snapshots. Furthermore, the initial value in IVPs usually is not of importance.

7.2.3 Evaluation

Let us now gather the previous findings as well as the results of the thesis in a final evaluation of the POD Method as a tool in Model Reduction and Suboptimal Control. Furthermore, we wish to point out possible problems of the method to be tackled in the future.

Trade-off between Speed and Quality By means of the findings above, let us denote possible choices of parameters in POD-ROM in order to tweak the "speed" or the "quality" of the method (refer also to "QC/SC" in Figure 7.2).

- A higher number of snapshots n increases the time needed to calculate a POD Basis, but also increases the amount of information captured in the snapshots, i.e., possibly increases the quality of the POD approximation. On the other hand, improving the *location* of snapshots may increase the quality of approximation *without* increasing the computation time.
- An accurate calculation of snapshots may improve the quality of the POD approximation but obviously takes more time.
- A larger number for the length of the POD Basis ℓ yields a higher dimensional ROM and hence leads to a longer solution time. On the other hand, it increases the amount of information in the snapshots captured by the POD Basis and hence may increase the quality of the POD approximation.
- Solving many "similar" systems with one ROM (implicitly) accelerates the calculation but decreases the quality of the POD approximation (for decreasing similarity).

How to Predict the "Quality" of a POD Reduced-Order Model? In order to judge on the actual quality of an approximation (or to even *predict* it), we need an estimation of the error in the approximation. We have seen, how to "trade" between speed and quality of a POD approximation – the respective choices are depicted in the right branch of Figure 7.2 as "QC/SC". Judging on the quality, we are interested in lower right node of the figure.
The left branch denotes the estimate which is covered by the estimates derived in Chapter 4. Therefore, we are interested in the assertions corresponding to the dashed arrows, i.e., the dependence of the "POD quality" on the "possible quality".
Basically, the quality of approximation depends on how well the actual characteristic of the system matches the "POD characteristics" found. In Figure 7.2, we see that this property depends on the actual "outcome of quality" in each of the choices "QC/SC". In particular, the following "disturbances" contribute to the error of the POD approximation and hence influence its quality (see also Figure 4.2):

- Perturbations in Snapshots
- Choice of snapshot sets (size and locations)
- Errors in the calculation of snapshots
- Reproducing only a certain amount of "energy" in the snapshots by a POD Basis.

Unfortunately, only the last point has sufficiently been dealt with: We have derived error estimates for the POD approximation of snapshots in contrast to the snapshots themselves (depending on the choice of ℓ).
All other aspects remain *open* at this stage and hence, we have *not* found a satisfying solution to the problem of "predicting the quality of a POD reduced-order model".

What to expect of the POD Method in Model Reduction? The construction of the procedure of POD Reduced-Order Modeling is somehow *paradox*: We wish to calculate a solution by means of a ROM, yet in order to setup this model we need to *know* the solution – at least at certain time instances ("snapshots"). Anyway, we could establish three procedures in which we benefit from applying the POD Method:

- In numerical examples, we found that POD ROM may be well-suited to *inter-* or *extrapolate* a given solution.
- There are chances in finding a satisfying low-order solution based on "perturbed" snapshot sets. We investigated how to judge on the quality of the approximation, but all these considerations remained very general. In particular, a corresponding actual error estimate is not yet established.

Chances in Suboptimal Control We proposed two methods of ensuring that the POD Basis models the characteristics of the optimal state ("basis update" and "OS-POD").
In general, *two* POD Basis have to be determined (one for the state and one for the adjoint state). In a practical calculation, there hence is quite a bit of time used on the *initialization* of the control algorithm: Snapshots have to be calculated for two systems and the *two* corresponding POD eigenvalue problems have to be solved.
Anyway, the POD Method in context suboptimal control certainly appears quite promising. (For example for fine discretizations in space – which increase the FE optimality system but do not significantly lengthen the calculation of a POD Basis.) Furthermore, there are many examples in which POD Suboptimal Control has been applied quite successfully.

Conclusion If the POD Method is applied correctly, it is quite likely to increase the speed of calculation. As seen above, estimating the quality of the low-order solution is a complex issue and far from actually achieved. The problem may probably only be tackled in a piece-wise fashion (i.e., for certain "types" of perturbations of snapshots).
To put it bluntly, the POD easily fastens a simulation, but to judge on its correctness is hard. In this sense, A. Chatterjee (unfortunately) was right in his utterance (2000):

> "The previous problems notwithstanding, through a combination of engineering judgment and *luck*, the POD continues to be fruitfully applied in a variety of engineering and scientific models."

7.3 Future work

We shall first present a collection of variants in using the POD Method which are carried out in the literature, yet we did not consider in this thesis. Then, let us give a brief "outlook" on what is to be done of the field of POD, particularly focusing on the drawback found above: the lack of estimates of the quality of a POD approximation.

7.3.1 Issues not Considered

Let us comment on aspects of the POD Method that have not been considered in this thesis in three different contexts.

Reduced-Order Modeling Surely, we could have considered other *types* of problems – such as "problems with dominant convection" (which would be interesting in context of problems with "traveling information"). Especially of interest would of course be *non-linear* problems. Furthermore, we could consider which difficulties *time-dependent* coefficients in the Evolution Problems present to the method.
In terms, of error estimates, we did not consider a time continuous error estimate which may be found in Henri 2003 for the space continuous case and in Volkwein 2006, Theorem 2.1 for the space discrete case.
As mentioned, there are quite a few other reduction methods which could be *compared* to the POD approach (Krylov based methods, for instance).
In the numerical calculations, we could have used more elaborated examples when (say) perturbing the snapshots (discontinuous examples, for instance).
Additionally, we could have numerically confirmed the error estimates established theoretically.

Suboptimal Control As far as problems to treat are concerned, there certainly is a vast variety of Optimal Control problems. In addition to other model choices there are alternative choices of the *type* of control – "optimal time control", for instance, where one tries to achieve a desired state as quick as possible.
In terms of treatment of problems, we could consider more sophisticated algorithms of solution (Newton-like methods as proposed). In a nonlinear case, we could consider SQP methods (refer to Volkwein 2001*a*, for example.)

Parameter Estimation The POD Method actually is not only applicable to *time*-dependent Evolution Problems but any parametrized set of data. (This should be obvious from the deduction of the theory from the general case where "time" was given the role of a *parameter* of the snapshots.) Therefore, we could also apply the method to (say) elliptic problems which depend on a parameter. The problem is then likely to become non-linear though. (Refer to the diploma thesis Kahlbacher 2006, for instance.)

7.3.2 Targeting the Problem of *"Unknown"* Quality

In our discussion, it has turned out that a major objective needs to be to establish reliable estimates to judge on the *quality* of a low-order solution.
Sources of errors in the practical process of ROM are depicted in Figure 4.2. Essentially, we need to take into account the four different choices for "QC/SC" in the process depicted in Figure 7.2 – solely the last one of them is actually understood (the influence of ℓ on the error of the POD approximation). Furthermore, the error estimates at the bottom of the diagram have been established.
Therefore, let us state objectives for the other three choices of "QC/SC".

Snapshot Perturbation – Estimates for Families of Problems? In a nutshell, we desire to find out: When are two systems "similar" enough such that the snapshots of the one may be used to construct a POD-ROM for the other?
More concretely, we ask for an error in the POD representation based on the perturbation in (say) individual parameters in a *family* of systems. (Particularly, we are interested in assertions of the following sort: "In this type of problem, the error in the POD representation is satisfyingly small as long as the parameter (say) c ranges over the interval (say) I.")

Optimal Snapshot Location – Capturing Characteristics in Snapshots The main question to pose is: Given a problem, how to choose a suitable snapshot grid? That implies: How to locate instances in time of "characteristic dynamics"?
A possible approach would be to introduce another optimization problem of the snapshot locations (in slight analogy to Gaussian quadrature formulas for numerical integration). On the one hand, it is likely to obtain an estimate for the *quality of snapshots* in this way. On the other hand, this additional problem would increase the computational effort, of course.

Asymptotic Behaviour of the POD Basis – Capturing Characteristics in the POD Basis
We are interested in how errors in the snapshots influence the POD Basis calculated from them – i.e., how those errors influence the "characteristics" found and hence, influence the error in the low-order solution. (The perturbation of the POD Basis in context of Suboptimal Control of parabolic problems was analyzed in Henri 2003, for example.)
Asymptotic estimates for the increasing number of (exact) snapshots are established. We also discussed *spatial* approximations of the snapshots (see also Volkwein 1999) which also yielded a *rate* of convergence. Actual estimates have however only been derived for the POD operator – it shall albeit be possible to derive estimates for the actual POD Basis (by means of perturbation theory of linear operators, for instance; refer to Kato 1980).
To the author's knowledge, the asymptotic behaviour of a POD Basis in terms of a *temporal* approximation of the snapshots is an open question. Estimates should be of help in answering (say): What is the difference of using snapshots of a coarse-grid-solution in contrast to using snapshots of a fine-grid-solution? The issue is also expected to be of some importance in context of analyzing POD as an *interpolation* or *extrapolation* scheme.

7.3.3 Other Improvements of the POD and Competing Methods

Of course, there are several ways to improve the POD Method. For example, J. Borggaard (Virginia Tech) gave a talk on "Improved POD: Parallel Algorithms and Basis Selection", in which he discussed "improvements to the way POD is carried out that are appropriate for complex, parameter dependent PDEs".
Let us give some further concepts of improvement of the method – all based on the idea of *linking* the two historical approaches to the POD Method. Finally, we shortly comment on a Model Reduction method which happens to be "better suited" than the POD Method in some situations (at present time).

Linking Approaches to POD As mentioned in Chapter A, there are (at least) two approaches to the POD Method: a *"statistical"* one from the point of view of "dynamical systems" and a *"numerical"* one from the point of view of "variational calculation" (variational formulations of Evolution Problems, for instance).
As seen in this thesis, the focus of these approaches do differ. Basically, in the first approach people are interested in the POD Basis *itself* as the objective of the approach usually is to *understand* the system at hand. In the latter approach, people wish to *use* the POD Basis (in a reduced-order model say) in order to calculate a solution of the system.
We sought to link these approaches and have established examples for either direction: The equivalence of two definitions of Coherent Structures was shown by virtue of a proof in Volkwein 1999. On the other hand, we understood the Method of Snapshots by means of the bi-orthogonal character of the POD Method.
Anyway, it is likely that there are more links to benefit from. Let us outline two of those in the remainder in order to encourage people to investigate them.

Use of a Specific "Direction" for Snapshots As snapshots, we so far have used *full solutions* to Evolution Problems at particular time instances. It might however be fruitful to concentrate on certain "directions" in space. In particular, one could make use of the degeneration of the POD Method to Fourier modes in some situations. (For a "structured approach to dimensions", refer to Holmes, Lumley, and Berkooz 1996, for instance.)

"Space Shot POD" and "Space-Time POD" Chances for carrying out the POD in order to determine "key *temporal* structures" have not been discussed. (They have only been used in the Method of Snapshots in order to calculate "key *spatial* structures" more easily.) In this approach, we would try to represent "snapshots" at certain coordinates in *space* – which we may refer to as *"space shots"*. The *correlation* of the "space shots" would be carried out in time whereas the *averaging* would take place in the space coordinate (which has in this case become the "ensemble parameter"). Corresponding reduced-order models would be of "horizontal" fashion and Suboptimal Control strategies could be similar to "instantaneous control" approaches (i.e., approaches in which the temporal dimension is "preferred").
Furthermore, we may investigate the possibilities of making use of both the orthogonal modes, i.e., applying POD in space *and* time. A good starting point is given in Volkwein and Weiland 2005, for example.

Concurrence for POD: Reduced *Basis* Approach In POD Suboptimal Control, the initial setup of snapshot sets and the calculation of one (or two) POD Basis significantly contribute to the total time of solution.
The main advantage of the "reduced *basis* approach" is that it (adaptively) calculates the actual basis of the low-order model directly. The resulting modes are not "orthogonal" at all, yet on the other hand, every solution calculated is actually used (As for the POD, only a few representatives of a genereally much larger "snapshot set" are used). Furthermore, time is saved since no additional basis calculation has to be carried out. Hence, the method turns out to be faster in some situations.
A "Reduced Basis Method for Control Problems governed by PDEs" may be found in Ito and Ravindran 1997, for instance.

Appendix A

Statistical Interpretation of the POD Method

In this chapter, we wish to investigate the POD Method from the viewpoint of statistics. In this way, we show how statistical concepts (in form of the POD) aid the numerical solution of Evolution Problems.
In particular, we apply notions from signal analysis (such as autocorrelation and signal decomposition). In this way, we obtain a better *understanding* of the properties of the method which have been of relevance in the numerical applications already: possible difficulties of the POD Method in the representation of snapshots as well as the alternative calculation of POD modes via the Method of Snapshots. Finally, we wish to apply the POD in order to find "Coherent Structures" in turbulent fluid flows.

Procedure Throughout, we assume to apply the POD Method to ensembles in an L^2-function space and setup a corresponding solution operator. By means of the notion of "autocorrelation", we give links of the POD to other approaches (say the Fourier decomposition) as well as hints on when the POD may work less nicely. By means of the additional "function space parameter", we introduce a "bi-orthogonal" decomposition in order to *interpret* the "Method of Snapshots". In this context, we illustrate in which way the "statistical" POD Method aides the numerical solution of Evolution Problems. We finally motivate the concept of Coherent Structures and give two definitions for which POD is capable of calculating such structures.

Historical Background – Motivation Note that the POD Method originally was only a *statistical* tool to extract "patterns" from a given data set. In particular, the POD was known to be a general theorem in probability theory (refer to Loève 1955). Only later, these patterns were used to construct low order models for actually *calculating* the dynamics of a system (as we have done in this thesis). In signal analysis (which we make use of in this chapter), the method is usually referred to as "Karhunen Loeve Decomposition" or "Principal Component Analysis".

A.1 Correlation Aspects of POD for Abstract Functions

In this section, we introduce the POD for ensembles of L^2-functions as well as the concept of "autocorrelation". We show its connection to the POD operator and draw a link to examples of snapshots set which the POD may struggle to represent.

Determining a POD Basis Let us consider an ensemble $\mathcal{V}$ in the Hilbert space $X := L^2(\Omega)$ of all square-integrable real valued functions, where Ω denotes a bounded domain in $\mathbb{R}^d, d \in \mathbb{N}$. In

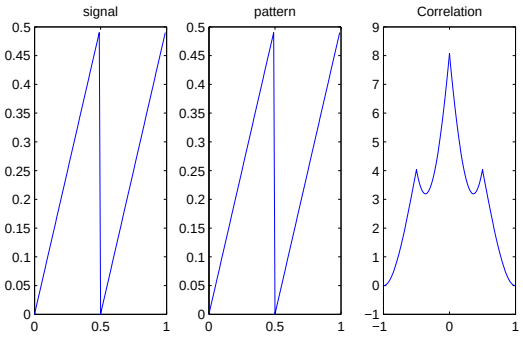

Figure A.1: Basic illustration of the concept of autocorrelation of a signal.

comparison to the abstract situation, we have only specialized the choice of the Hilbert Space X. Hence, all we need to specify is a suitable inner product – which we choose to be the standard one:

$$(v, w)_{L^2(\Omega)} := \int_\Omega v(x)w(x)\,\mathrm{d}x.$$

The problem statement coincides with the one of Definition 2.1.5 for $X := L^2(\Omega)$. Analogously, we may determine the characterizing POD operator from the operator of Theorem 2.2.3 and state this as a Corollary. (Note that we now have $y(t) \in L^2(\Omega)$ for each $t \in \Gamma$. Thus, we may denote the real value of $y(t)$ at $x \in \Omega$ by $y(t)(x)$.)

Corollary A.1.1 (Solution of POD Problem in $L^2(\Omega)$)
Define the POD operator $R_L : L^2(\Omega) \to L^2(\Omega)$ by

$$(R_L\psi)(x) = \Big\langle y(t)(x)\,(y(t), \psi)_{L^2(\Omega)} \Big\rangle_{t\in\Gamma} = \left\langle y(t)(x) \int_\Omega y(t)(z)\psi(z)\,\mathrm{d}z \right\rangle_{t\in\Gamma}.$$

Let $\{\lambda_k\}_{k\in\mathbb{N}}$ be a (decreasingly) ordered set of eigenvalues and $\mathcal{B} = \{\lambda_k\}_{k\in\mathbb{N}}$ an orthonormal set of corresponding eigenvectors of R_L such that $\mathcal{B}$ denotes a basis of $\mathcal{V}$. Then, $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ (i.e. an orthonormal set of eigenvectors of R_L corresponding to the ℓ first (largest) eigenvalues) denotes a POD Basis of rank ℓ.

Proof.
Comparing R_L to the definition of the abstract operator R in (2.5), the assertion may be easily derived from Theorem 2.2.3. A proof is also given in Holmes, Lumley, and Berkooz 1996. A proof for a special choice of average operator may be found in Volkwein 2001*b*, Example 3. □

Autocorrelation Autocorrelation (AC) is a concept to identify *repeating patterns* in a signal or finding a periodic signal which has been buried under noise.
More strictly speaking, the autocorrelation function $AC(\tau)$ measures how well a signal matches a shifted version of itself – depending on the amount of shift τ. Note that a basic example is depicted in Figure A.1. In particular, the (averaged) autocorrelation function AC_f of a suitable signal f (for $\langle\cdot\rangle_{t\in\Gamma} := \frac{1}{T}\int_0^T \cdot\,\mathrm{d}t$) is defined as

$$AC_f(\tau) := \langle f(t+\tau)f(t)\rangle_{t\in\Gamma} = \frac{1}{T}\int_0^T f(t+\tau)f(t)\,\mathrm{d}t. \tag{A.1}$$

Further Interpretation of the POD – Autocorrelation Let us now draw a connection of the POD to autocorrelation by "interpreting" the POD operator. In particular, we introduce a kernel r which is of the form of an autocorrelation (defined in (A.1)).

Corollary A.1.2 (Autocorrelation Property)
The POD operator R_L is an integral operator whose kernel may be represented by an *averaged autocorrelation* function $r : \Omega \times \Omega \to \mathbb{R}$ (w.r.t. the snapshots). In particular, for $\psi \in L^2(\Omega)$, the operator R_L may be written as

$$(R_L\psi)(x) = \int_\Omega \psi(z)\, r(x,z)\, \mathrm{d}z \quad \text{with} \quad r(x,z) := \langle\, y(t)(x)\, y(t)(z)\,\rangle_{t\in\Gamma}\,. \tag{A.2}$$

Proof.
Recall that in the definition of a POD Problem, the average operation was assumed to commute with the inner product. We may thus obtain the assertion by swapping the average operation and the integration in the definition of the operator R_L in Corollary A.1.1. □

Decomposition of Autocorrelation We have seen that the kernel of the POD operator is given by an autocorrelation operator. We now wish to show that the POD modes actually decompose this autocorrelation operator. (This result will enable us to prove that Coherent Structures (in the sense proposed by Sirovich) may be obtained by POD modes.)
We prove this result in the fashion proposed in Volkwein 1999. Note however that the assertion may also be obtained by means of "functional analysis" – in particular, the so called *Mercer's Theorem.*

Proposition A.1.3 (Decomposition of the Autocorrelation Operator)
Let $\mathcal{B}^\ell = \{\psi_k\}_{k=1}^\ell$ denote a POD Basis determined by R_L. Let r be the kernel of R_L in the sense of Corollary A.1.2. Then, there holds

$$r(x,z) = \sum_{k\in\mathbb{N}} \lambda_k \psi_k(x)\psi_k(z). \tag{A.3}$$

Proof.
Restating the denition of R_L in (A.2), we have

$$(R_L\psi)(x) = \big(\psi(\cdot), r(x,\cdot)\big)_{L^2(\Omega)}. \tag{A.4}$$

Obviously, for every fixed $x \in \Omega$, we see

$$r(x,\cdot) = \langle\, y(t)(x)\, y(t)(\cdot)\,\rangle_{t\in\Gamma} \in \mathcal{V} = \operatorname{span}(\mathcal{V}_P),$$

since $r(x,\cdot)$ basically denotes a weighted average of elements $y(t)(\cdot) \in \mathcal{V}_P$.
Furthermore, $\{\psi_k\}_{k\in\mathbb{N}}$ denotes an orthonormal basis for $\mathcal{V}$. Hence, we may represent $r(x,\cdot)$ in terms of this basis, use equation (A.4) and finally use the fact that (for all $k \in \mathbb{N}$) ψ_k is an eigenvectors of R_L. For arbitrary $z \in \Omega$, we in this way find the assertion:

$$\begin{aligned} r(x,z) &= \sum_{k\in\mathbb{N}} \big(\psi_k(\cdot), r(x,\cdot)\big)_{L^2(\Omega)}\, \psi_k(z) = \sum_{k\in\mathbb{N}} (R_L\psi_k)(x)\psi_k(z) \\ &= \sum_{k\in\mathbb{N}} \lambda_k \psi_k(x)\psi_k(z). \end{aligned}$$

□

Symmetry of Autocorrelation and Fourier Degeneration Quite a few "symmetries" may be helpful when applying the POD Method. For example, in Holmes, Lumley, and Berkooz 1996, Subsection 3.3.3 it is discussed to *extend* the snapshot set by means of symmetries.
Let us focus on a special kind of symmetry: homogeneity (in certain directions). In particular, let us show that the POD Method "degenerates" to a Fourier decomposition in homogeneous directions of the autocorrelation operator r. (This may especially speed up the calculating a POD Basis for problems in higher dimensions.)

Proposition A.1.4 (Degeneration to Fourier Series)
The POD Method degenerates to a Fourier expansion in space (i.e., the eigenvectors of the operator R_L are Fourier modes) if and only if the *averaged two point correlation* r of (A.2) is *translation invariant*, i.e., if there holds $r(x,z) = r(x-z)$ in some direction. (This property is also called "homogeneity".)

Proof.
We give a sketch of the proof. For more details refer to Holmes, Lumley, and Berkooz 1996, Subsection 3.3.3.
We assume that r is homogeneous. In the case of a finite domain we may develop $r(x-x')$ into a Fourier series and obtain:

$$r(x-x') = \sum_{k\in\mathbb{N}} c_k e^{2\pi i k(x-x')} = \sum_{k\in\mathbb{N}} c_k e^{2\pi i k x} e^{-2\pi i k x'} = r(x,x'). \tag{A.5}$$

Inserting this ansatz into the eigenvalue problem for R_L, we see that $\{e^{2\pi i k x}\}$ are exactly the eigenfunctions with eigenvalues c_k, i.e., denote the POD modes.
Conversely, if the eigenfunctions for R_L are Fourier modes, we may read (A.5) from right to left and obtain the homogeneity of the auto-correlation operator r. □

POD and Correlations Analogously to the operator R, let us now deduce a "kernel" for the operator K, which is of "averaged autocorrelation form" (for $X := L^2(\Omega)$). Let us additionally comment on the *type* of autocorrelation involved – for K as well as for R.

Corollary A.1.5 (POD Correlation Kernels)
Let y denote the parametrization of an ensemble over Γ. (I.e., for $t \in \Gamma$ there holds $y(t) \in L^2(\Omega)$ and hence it makes sense to write $y(t)(x)$).
Then, there are representations of the kernel k of the operator K as well as of the kernel r of R which are of *autocorrelation form*. In particular, for all $v \in L^2(\Gamma)$ we find

$$(Kv)(t) = \int_0^T k(t,s)v(s)\,\mathrm{d}s \quad \text{with} \quad k(t,s) := \int_\Omega y(t)(x)\, y(s)(x)\,\mathrm{d}x \tag{A.6}$$

$$\text{and} \quad r(x,z) = \int_\Gamma y(t)(x)\, y(t)(z)\,\mathrm{d}t. \tag{A.7}$$

Since each "ensemble member" is an L^2-function, we furthermore deduce that the kernels correspond to each other in the following sense:

- The kernel k is an autocorrelation operator that measures the self-correlation of values of "ensemble members" for different shifts in the ensemble parameter (averaged over the argument in each ensemble member).
- The kernel r measures the self-correlation of values of "ensemble members" for different shifts in their argument (averaged over the ensemble parameter).

Proof.
Actually, we had stated the operator K in Proposition 2.2.10 in a "swapped" way already such that we may directly infer the definition (A.6) of the kernel k from (2.19):

$$(Kv)(t) = \int_0^T (y(t), y(s))_{L^2(\Omega)}\, v(s)\,\mathrm{d}s = \int_0^T k(t,s)v(s)\,\mathrm{d}s$$
$$\text{with} \quad k(t,s) := \int_\Omega y(t)(x)\, y(s)(x)\ \mathrm{d}x.$$

In order to prove the assertion (A.7) on r, we make use of its definition in (A.2). Due to the definition of the average operator, we directly infer:

$$r(x,z) := \langle\, y(x)\, y(z)\,\rangle_{t\in\Gamma} = \int_\Gamma y(t)(x)\, y(t)(z)\ \mathrm{d}t.$$

□

Implications for the Capabilities of the POD Corollary A.1.5 teaches us that both the POD operators essentially are determined by an "autocorrelation of the data in the ensemble". Therefore, we expect the POD Method to struggle in case that there is not much "correlation" between either the ensemble members or the data within each ensemble member.
We have practically experienced such difficulties in Chapter 6 for the "challenging" example. In this problem, the information "traveled quickly" in time as well as in space. Therefore, the correlation of the data is *little* – no matter whether we choose the time or the space variable as an ensemble parameter (i.e., whether we choose to calculate the POD Basis via R or K).
It would be interesting in this context to investigate problems whose speed of information is asymmetric in time and space. We could then investigate if we could improve the POD Basis by choosing the "suitable" kernel, depending on whether the information travels quickly w.r.t. the ensemble parameter or w.r.t. the argument in the ensemble members.

A.2 Bi-Orthogonal Decomposition of Signals of Snapshots

In this section, we symmetrize the parametrization of the ensemble in order to represent the data in the ensemble by a "signal". We show how to "bi-orthogonally" decompose this signal and give a *statistical characterization* of the POD Method.

Procedure We introduce the POD Method for ensembles of square-integrable functions. We show how this ensemble may be interpreted as a "space-time-signal" – of which we establish a "bi-orthogonal decomposition". Then, we explain that the POD Method actually leads to this decomposition, too. By means of this, we "illustrate" the Method of Snapshots. Finally, we comment on the "statistical" in contrast to the "numerical" aspects of the POD.

A.2.1 A "Signal" of POD Ensemble Members

We explain in how far we deal with an ensemble of data that depends on two variables. We introduce the notion of a signal as a parametrization that treats both the parameters "equally".

An Ensemble of L^2-Functions By means of the POD, we desire to represent an ensemble $\mathcal{V}_P$ which is parametrized over a set Γ. We chose to represent an ensemble of L^2-functions which we

denote by $\mathcal{V}_P = \{y(t) \in L^2(\Omega) \mid t \in \Gamma\}$. We may interpret this to represent *real data* which depends on an additional parameter $x \in \Omega$.
Altogether, our ensemble in this sense consists of real data which depends on *two* parameters – the ensemble parameter t as well as the parameter x. Alternatively, we may say that our ensemble is parametrized by an *abstract function* $G \in L^2(\Gamma; L^2(\Omega))$ (refer to Chapter 1).
So far, this *asymmetric* treatment of parameters was well suited to the fact that we employed the "vertical" approach for Evolution Problems: We took snapshots in time in order to extract "key *spatial* ingredients" by means of which we setup a time-continuous model. I.e., we chose the "time" to play the role of an ensemble parameter and the "space" to become the parameter in *each* "ensemble member".

Definition of a "Signal" In general, it does not matter which parameter of the "data" in the ensemble is preferred, i.e., becomes the *"ensemble parameter"*. Therefore, we now wish to parametrize the ensemble in the two parameters *symmetrically*. For this purpose, we define the notion of a *"signal"* which in contrast to abstract functions depends on both the parameters without preference. In particular, we set:

$$u \in L^2(\Omega \times \Gamma), \quad u(x,t) := y(t)(x). \tag{A.8}$$

Note that this roughly corresponds to the usual identification of $L^2(\Gamma; L^2(\Omega)) \cong L^2(\Omega \times \Gamma)$.
In this sense, we may now say that we wish to represent a signal $u \in L^2(\Omega \times \Gamma)$, where the "order" of variables is not determined. Without loss of generality, let us conveniently assign the *roles* of time and space to them. (The parameters are not distinguished anymore.)

A.2.2 Bi-Orthogonal Decomposition of Signals

We introduce a "bi-orthogonal decomposition of signals", comment on the properties of its ingredients and introduce a way to calculate them in order to draw links to the POD.

The Basic Theorem Let us define a "bi-orthogonal decomposition of signals" (and state its existence) in the following theorem.

Theorem A.2.1 (Existence of a Bi-Orthogonal Decomposition)
Let $u \in L^2(\Omega \times \Gamma)$ be a signal. Then, there exist *"spatial modes"* $\{\psi_k\}_{k\in\mathbb{N}} \subset L^2(\Omega)$ and *"temporal modes"* $\{\varphi_k(t)\}_{k\in\mathbb{N}} \subset L^2(\Gamma)$ such that:

$$u(x,t) = \sum_{k=1}^{\infty} \alpha_k \psi_k(x) \varphi_k(t)$$

with

$$\alpha_1 \geq \alpha_2 \geq \cdots > 0 \quad \text{and} \quad \lim_{k\to\infty} \alpha_k = 0.$$

Furthermore, the spatial as well as the temporal modes are *orthonormal*:

$$(\psi_i, \psi_k)_{L^2(\Omega)} = (\varphi_i, \varphi_k)_{L^2(\Gamma)} = \delta_{ik}.$$

Proof.
Refer to Aubry, Guyonnet, and Lima 1991, Theorem 1.5. □

Review of Decomposition Let us explicitly denote the properties of the bi-orthogonal decomposition which are implied by Theorem A.2.1. In particular, we wish to point out that we may carry out a *Model Reduction* by "sorting" the contributions of modes and "truncating" their sum.

- The spatial as well as the temporal modes are *"uncorrelated"* ("orthogonal"). (This fact obviously has given the method its name.)
- The modes are coupled: Each spatial mode is associated with a temporal "partner": The latter is the time evolution of the former.
- The "space-time-partners" are *ordered* by their "contribution" to the signal. (The modes are normalized and the respective coefficients decrease.)
- An "optimal" representation of the signal by ℓ modes is given by the first ℓ modes of the decomposition. (Since they are ordered by their contribution.)

Calculation of Bi-Orthogonal Modes In analogy to the POD Method, let us "characterize" the (spatial and temporal) modes as eigenvectors of suitable operators. We base these characterizations on an operator that maps time modes on space modes and vice versa (analogously to the decomposition of the operator R in Proposition 2.2.7).

Proposition A.2.2 (Calculation of Bi-Orthogonal Modes)
Set $X := L^2(\Omega)$ and $T := L^2(\Gamma)$. Introduce the operator

$$Y : T \to X, \quad (Y\varphi)(x) = (u(x,\cdot),\varphi)_T = \int_\Gamma u(x,t)\varphi(t)\,\mathrm{d}t \quad \text{for } \varphi \in T$$

as well as its adjoint

$$Y^* : X \to T, \quad (Y^*\psi)(t) = (u(\cdot,t),\varphi)_X \quad \text{for } \psi \in X.$$

Then, we find that the "spatial modes" $\{\psi_k\}_{k\in\mathbb{N}} \subset L^2(\Omega)$ and "temporal modes" $\{\varphi_k(t)\}_{k\in\mathbb{N}} \subset L^2(\Gamma)$ are eigenvectors of $R := YY^*$ and $K := Y^*Y$, respectively:

$$R\psi_k = YY^*\psi_k = \alpha_k^2\psi_k \quad \text{and} \quad K\varphi_k = Y^*Y\varphi_k = \alpha_k^2\varphi_k.$$

Furthermore, we may "convert" the modes into each other by means of the operator Y as well as its adjoint:

$$\psi_k = \alpha_k^{-1}Y\varphi_k \quad \text{and} \quad \varphi_k = \alpha_k^{-1}Y^*\psi_k.$$

Proof.
The spatial modes are eigenvectors of the operator R due to Aubry, Guyonnet, and Lima 1991, Proposition 1.7. All other assertions are justified within the proof of Aubry, Guyonnet, and Lima 1991, Theorem 1.5. □

A.2.3 Application to the POD Method

Let us now apply our findings for the bi-orthogonal decomposition to the POD Method. Together with the aspects of autocorrelation, we shall then characterize the POD Method as a "statistical" tool.

Link to POD Note that the definitions of the operators Y and Y^* in Proposition A.2.2 *"coincide"* with the respective definitions in Proposition 2.2.7.
Therefore, we find that the spatial modes ψ in the "bi-orthogonal decomposition" are POD modes (since both of them are eigenvectors of $R = YY^*$).

Relation to the POD in Model Reduction In this thesis, we chose to discretize the Evolution Problem in the "vertical" way: via a discretization in space, we obtained a (time-continuous) system of equations. In order to reduce this system, we wanted to setup a reduced-order model consisting only of the key *spatial* patterns of the solution. According to the findings above, such key spatial structures are given by the POD modes (at least in terms of snapshots).
Furthermore, note that the POD Method in certain circumstances coincides with the method of "Balanced Truncation". (We have seen "Model Reduction by truncation" in the itemized list above already.)

Understanding the Method of Snapshots We have seen that the POD Method yields the key *spatial* structures in a snapshot set which are needed in order to reduce the order of the *spatial* discretization of (say) Evolution Problems.
On the other hand, according to Corollary A.2.2, in the Method of Snapshots we actually calculate key *temporal* structures φ (since we calculate eigenvectors of K – see Theorem 3.1.5).
We may now understand that in the second step in the Method of Snapshots we "transform" the temporal modes calculated into the (spatial) POD modes by means of the *coupling* between the temporal and spatial mode pairs. (In context of an "asymmetric" treatment of the parameters in the snapshot set, we may say that we *act* as if the snapshots were parametrized in space.)

Theoretical Study of Mean Reduction In Subsection 6.1.4, we numerically studied the influence of the mean subtraction in the snapshot set. Let us mention that a theoretical investigation in context of bi-orthogonal decompositions is carried out in Aubry, Guyonnet, and Lima 1991, Theorem 1.13 and Remark 1.14.

Connection to SVD Note that on a matrix level, the bi-orthogonal decomposition is simply given by an SVD of the ensemble matrix. The (right and left) singular vectors are orthonormal and coupled in a suitable fashion. Furthermore, they are "ordered" by the corresponding singular values. Therefore, we see that on this "matrix level" the bi-orthogonal decomposition introduced above may be given by an SVD.

Statistical Characterization of the POD By means of Proposition A.2.2, we have seen that POD modes represent key *spatial* modes in the snapshot set. According to Proposition A.1.3, the POD modes decompose the kernel r and therefore also represent the essential "ingredients" of the *spatial* correlation operator averaged over time.
Therefore, we may conclude that the POD Method finds key *spatial* structures by a *correlation* of the "signal of snapshots" with a *spatially* shifted version of itself and *averaging* this quantity over *time*.

Summary: POD as a Statistical Model Reduction Altogether, we see: The *statistical* concepts of "average" and "correlation" determine key structures that aid the "numerical process" of solving (say) an Evolution Problem.
In particular, for our vertical approach, we identify key *spatial* structures by averaging in time and correlating in space. Alternatively, we may average in space and correlate in time in order to determine key *temporal* structures. We may then convert these modes to the key *spatial* structures of desire by means of the *statistical* decomposition of signals.
We have illustrated this interaction of "numerics" and "statistics" in Figure A.2.

A.3 Identification of Coherent Structures in Turbulent Flows

In this final section, let us investigate in which sense the POD Method aids a process that is "off the track" of the remainder of the thesis: Basically, we aim to find "underlying patterns" in the a priori "chaotic" appearance of a flow in turbulent state – so called *Coherent Structures*.

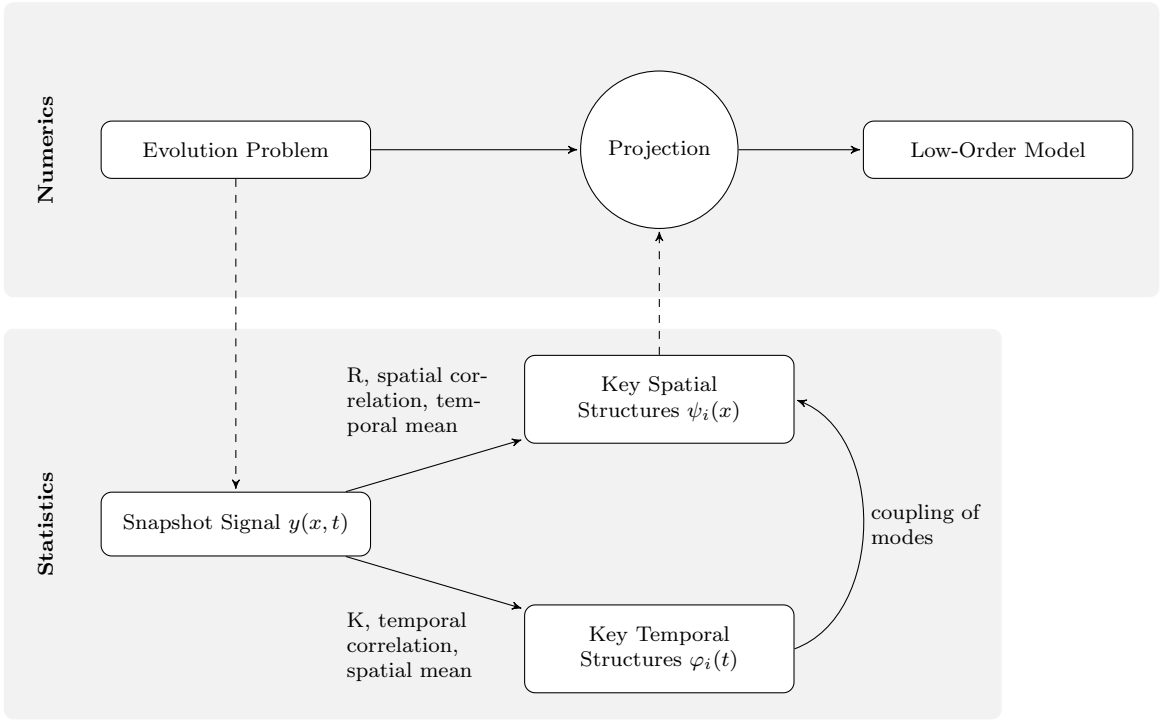

Figure A.2: The POD Method as a *statistical* approach to setup reduced-order models that shall fasten *numerical* simulations.

Terminology: Dynamical Systems Throughout the thesis, we have referred to models which determine a temporal behaviour of a system as "Evolution Problems". In this context, we shall use the term "dynamical systems" in order to stress the difference in the focus of investigation. In terms of "dynamical systems", this focus is somewhat more "global", the questions posed are more "general": Which invariant subspaces occur? Which "attractors" may be identified and of which dimension are they? Note that such problems may already be challenging in *two* dimensional dynamical systems.

Procedure We introduce the matter of turbulence as well as implied numerical problems. We explain the objectives of Coherent Structures as well as the idea of "Reduction of Turbulence". For two concrete definitions, we show that coherent Structures may be determined by the POD Method.

A.3.1 The Challenge of Turbulence

Let us briefly introduce the issue of "turbulence" in a flow and comment on the resulting numerical challenges. Some of these shall be helped by the concept of "Coherent Structures".

The Notion of Turbulence Turbulent (say) fluid flows differ from "laminar" flows in their *chaotic* behaviour as well as their "cascade of energy" up to a generally small scale ("Richard's model"). This ("Kolmogorov length") scale is determined by the so-called "Reynolds number" which determines the overall behaviour of a flow.

Numerical Problems in Turbulence Due to several reasons, tackling turbulent fluid flows numerically is very challenging

- There are only very few "explicit" solutions to the most common model for the simulation of flows, i.e., to the "Navier Stokes Equations".

- Linearizations of the *non-linear* model destroy the effects of turbulence.
- The simulation of the flow has to be carried out in three space dimension since the effects of turbulence are significantly different in two space dimensions.
- The problem generally is ill-posed in the sense that the solution critically depends on the initial- as well as on the boundary conditions ("Icy wings and turbulent airflow could cause a plane to crash.")
- In an FE simulation, a very high number of very small cells is needed in order to capture also the smallest eddies within a "large" domain ("Richardson's model" of scales).
- Omitting small contributions to the flow is difficult since their influence on the actually interesting scales is not understood. (The proper "characterization of the interaction of different scales" is an open problem.)

A.3.2 Introduction to Coherent Structures

In this subsection, we shall introduce the basic idea of Coherent Structures. As before, we are interested in making (numerical) problems feasible. In this context however, the resulting low-order model generally is chaotic and is hence to be investigated by the theory of dynamical systems. Therefore, the "order" of the low-dimensional model has to be even lower than before.

Objectives of Coherent Structures From a numerical analysis point of view, Coherent Structures shall contribute to more accurate as well as "easier" modeling of turbulent flows. For example, they should give hints on the influence of small scales on large scales and hence on estimating the error when ignoring small scales in numerical computations.
From a more general point of view, Coherent Structures shall help examining the dynamics of the turbulent flow and therefore aid understanding the "nature of turbulence" better: investigating the transfer processes of momentum, heat and mass, for example.

Idea: "Reduction of Turbulence" A turbulent flow is disordered in space, time or both. There are many unsteady vortices on many scales and it is hard to predict the spatial- as well as the temporal structure. – Such "chaotic systems" are studied in dynamical system theory. Unfortunately, the theory generally is only well developed for *very* low dimensional systems. Therefore, we want to understand turbulent dynamics on *low dimensional* subspaces. Basically, we wish to "restrict" the "turbulent behaviour" to these subspaces and, in this sense, "reduce" turbulence.
In order to establish such subspaces, we *assume* that turbulence consists of *coherent motions* and *incoherent (random) motions*, which are superimposed and may extend beyond the respective domains of coherent motions. We then try to decompose given flow data into these "deterministic motions" and random coefficients (*"deterministic chaos"*). The problem is then reduced to investigating these (hopefully few) coefficients. (This approach overall is similar to a bi-orthogonal decomposition, yet for this, we had assumed both components to be deterministic.)

Coherent Structures In terms of Coherent Structures, we explicitly choose to decompose space-time flow data (experimental or simulated) into "deterministic spatial functions" and "random time coefficients":

$$y(x,t) = \sum_k \text{ "coherent"}_k(x) \ \text{ "random"}_k(t). \tag{A.9}$$

In particular, we look for coherent *spatial* structures of *extension* in space, *persistence* in time or *significance* in dynamics. We aim to find such structures by (statistical) pattern recognition techniques (such as the autocorrelation).
In this sense, *Coherent Structures* are a concept which merges numerical as well as statistical approaches: By means of "statistical techniques" information is extracted from "numerical simulations".

A.3.3 Coherent Structures and POD

We select two of many types of Coherent Structures and show that these may be calculated by means of the POD Method. (Other definitions as well as a thorough discussion of the concept of Coherent Structures are given in Hussain 1983, for instance.)

Lumley/Sirovich Let us give definitions of Coherent Structures that are supposed to be very efficient as a basis of fluid data in a decomposition of the type (A.9). (Note however that these sorts of Coherent Structures are not necessarily observed in the fluid flow, but are only "persistent" structures in a statistical sense: "On average they are there.")

Definition A.3.1 (Coherent Structures according to Lumley/Sirovich)
Within fluid data (obtained by either experiment or simulation), a coherent structure is, according to

- Lumley, a mode with *high energy* contribution.
- Sirovich, an element in the diagonal decomposition of the averaged autocorrelation of the fluid data. I.e., Coherent Structures are *main contributions* to the representation of the *autocorrelation* of the "fluid data signal".

Calculation by POD We may easily conclude that the types of Coherent Structures introduced may be obtained by the POD Method.

Proposition A.3.2 (Use of POD in Context of Coherent Structures)
The Coherent Structures in the sense of Lumley as well as of Sirovich may be obtained by the POD Method. Therefore, the two definitions (technically) coincide.

Proof.
If we use the POD Method with L^2-Optimality Norm, for each possible dimension ℓ, the POD modes are the contribution of highest energy (refer to Subsection 4.3.2).
On the other hand, Proposition A.1.3 teaches us that the POD modes also decompose the autocorrelation operator of the signal of concern. □

Generalized Characterization of the POD Method Taking up on the statistical characterization of the POD Method in Subsection A.2.3 as well as on Proposition A.3.2, we understand the following characterization of the POD Method in context of turbulence (given in the abstract of Aubry 1991): "The [proper orthogonal] decomposition extracts deterministic functions from second-order statistics [(autocorrelation)] of a random field and converges optimally fast in quadratic mean (i.e., in mean energy)."

Model Reduction vs Interpreting POD Modes We have seen that Coherent Structures are used as a Model Reduction technique ("Reduction of Turbulence"). Note however that the structures *themselves* (in our case the POD modes) may be of interest in order to better *understand* the notion of turbulence (see "Objectives of Coherent Structures" above).

Appendix B

References

Chapter 1: Mathematical Basics

Alt, H. W. (1992). *Lineare Funktional Analysis*. Springer-Verlag, Berlin [u.a]. see p. 13.

Antoulas, A. (2005). *Approximation of Large-Scale Dynamical Systems*. Society for Industrial and Applied Mathematics. see pp. 10, 12, 60.

Brenner, S.C. and L.R. Scott (2002). *The Mathematical Theory of Finite Element Methods*. Springer-Verlag, Berlin [u.a]. see p. 24.

Dautray, R. and J.-L. Lions (1992). *Mathematical Analysis and Numerical Methods for Science and Technology, Volume 5: Evolution Problems I*. Springer-Verlag, Berlin [u.a]. see p. 18.

Dobrowolski, M. (2006). *Angewandte Funktionalanalysis*. Springer-Verlag, Berlin [u.a]. see p. 13.

Knabner, P. and L. Angermann (2000). *Numerik partieller Differentialgleichungen: Eine anwendungsorientierte Einführung*. Springer-Verlag, Berlin [u.a]. see pp. 14, 24.

Lube, G. (2005). "Lineare Funktionalanalysis." see pp. 12, 29.

— (2007). "Theorie und Numerik instationärer Probleme." see pp. 15, 16, 18–21.

Stewart, G. W. (1973). *Introduction to Matrix Computations*. ed. by W. Rheinbold. Academic Press, Inc. see pp. 10–12.

Zeidler, E. (1990). *Nonlinear Functional Analysis and its Applications*. Springer-Verlag, Berlin [u.a]. see p. 15.

Chapter 2: The POD Method in Hilbert Spaces

Holmes, P. J. L. Lumley, and G. Berkooz (1996). *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press. see pp. 27, 32, 133, 139, 142, 144.

Kato, T. (1980). *Perturbation Theory for Linear Operators*. Springer-Verlag, Berlin [u.a]. see pp. 39, 44, 138.

Kunisch, K. and S. Volkwein (2002). "Galerkin POD Methods for a General Equation in Fluid Dynamics." in: *SIAM Journal Numerical Analysis* 40.2, pp. 492–515. see pp. 8, 27, 41, 45, 54, 59, 63, 66, 70, 76, 77.

— (2006). *POD for Optimality Systems*. tech. rep. IMA10-06. University of Graz. see pp. 40, 101.

Lube, G. (2005). "Lineare Funktionalanalysis." see pp. 12, 29.

Reed, M. and B. Simon (1980). *Methods of Modern Mathematical Physics. I: Functional Analysis*. Academic Press. see p. 37.

Volkwein, S. (2001*b*). "Optimal Control of a Phase-Field Model Using POD." in: *Zeitschrift für Angewandte Mathematik und Mechanik* 81.2, pp. 83–97. see pp. 27, 33, 35, 42, 142.

— (2006). "Model Reduction Using POD." Lecture Notes. see pp. 8, 35, 47, 48, 65, 71, 78, 86, 101, 103, 104, 137.

Chapter 3: The POD Method for Evolution Problems

Chatelin, F. (1983). *Spectral Approximation of Linear Operators*. Academic Press Inc., New York. see p. 58.

Chatterjee, A. (2000). "An Introduction to the Proper Orthogonal Decomposition." in: *Current Science* 78.7. see pp. 53, 59, 81.

Demmel, J. W. (1997). *Applied Numerical Linear Algebra*. SIAM Philadelphia. see p. 58.

Kahlbacher, M. (2006). "POD for Parameter Estimation of Bilinear Elliptic Problems." MA thesis. University of Graz. see pp. 47, 55, 137.

Kunisch, K. and S. Volkwein (1999). "Control of the Burgers Equation by a Reduced-Order Approach Using POD." in: *Journal of Optimization Theory and Applications* 102.2, pp. 345–371. see p. 48.

— (2001). "Galerkin POD Methods for Parabolic Problems." in: *Numerische Mathematik* 90, pp. 117–148. see pp. 53, 59, 62, 63, 77.

— (2002). "Galerkin POD Methods for a General Equation in Fluid Dynamics." in: *SIAM Journal Numerical Analysis* 40.2, pp. 492–515. see pp. 8, 27, 41, 45, 54, 59, 63, 66, 70, 76, 77.

Sirovich, L. (1987). "Turbulence and the Dynamics of Coherent Structures." in: *Quarterly of Applied Mathematics* XLV, 3, pp. 561–590. see p. 54.

Volkwein, S. (1999). "POD and SVD." see pp. 8, 47, 58, 80, 138, 139, 143.

— (2006). "Model Reduction Using POD." Lecture Notes. see pp. 8, 35, 47, 48, 65, 71, 78, 86, 101, 103, 104, 137.

Chapter 4: Reduced Order Modeling for Evolution Problems

Antoulas, A. (2005). *Approximation of Large-Scale Dynamical Systems*. Society for Industrial and Applied Mathematics. see pp. 10, 12, 60.

Chatterjee, A. (2000). "An Introduction to the Proper Orthogonal Decomposition." in: *Current Science* 78.7. see pp. 53, 59, 81.

Hinze, M. and S. Volkwein (2004). "POD Surrogate Models for Nonlinear Dynamical Systems: Error Estimates and Suboptimal Control." SFB609 Preprint 27-2004. see pp. 78, 83, 100.

— (2005). *Error Estimates for Abstract Linear-Quadratic Optimal Control Problems Using POD*. tech. rep. 2/2005. Institutes for Mathematics (Graz University of Technology) & Institutes for Mathematics and Scientific Computing (University of Graz). see pp. 61, 83, 86, 98, 99.

Hömberg, D. and S. Volkwein (2003). "Control of Laser Surface Hardening by a Reduced-Order Approach Using POD." in: *Mathematical and Computer Modelling* 38, pp. 1003–1028. see pp. 75, 78.

Kunisch, K. and S. Volkwein (2001). "Galerkin POD Methods for Parabolic Problems." in: *Numerische Mathematik* 90, pp. 117–148. see pp. 53, 59, 62, 63, 77.

— (2002). "Galerkin POD Methods for a General Equation in Fluid Dynamics." in: *SIAM Journal Numerical Analysis* 40.2, pp. 492–515. see pp. 8, 27, 41, 45, 54, 59, 63, 66, 70, 76, 77.

Volkwein, S. (1999). "POD and SVD." see pp. 8, 47, 58, 80, 138, 139, 143.

— (2006). “Model Reduction Using POD.” Lecture Notes. see pp. 8, 35, 47, 48, 65, 71, 78, 86, 101, 103, 104, 137.

Chapter 5: (Sub) Optimal Control of Evolution Problems

Afanasiev, K. (2002). “Stabilitätsanalyse, niedrigdimensionale Modellierung und optimale Kontrolle der Kreiszylinderumströmung.” PhD thesis. Technische Universität Dresden; Fakultät für Maschinenwesen. see p. 100.

Atwell, J. A. and B. B. King (1998). “POD for Reduced Basis Feedback Controllers for Parabolic Equations.” in: *Math. and Comput. Model.* see p. 102.

Benner, P. (2001). “Numerical Solution of Optimal Control Problems for Parabolic Systems.” in: *SIAM-EMS Conference Berlin.* see p. 104.

Benner, P. S. Goerner, and J. Saak (2006). “Numerical Solution of Optimal Control Problems for Parabolic Systems.” in: *Lecture Notes in Computational Science and Engineering* 52. in K.H. Hoffmann and A. Meyer: Parallel Algorithms and Cluster Computing. Implementations, Algorithms and Applications. pp. 151–169. see pp. 101, 103, 104.

Bergmann, M. (2004). “Optimisation Aérodynamique par Réduction de Modèle POD et Contrôle Optimal. Application au Sillage Lamin d’un Cylindre Circulaire.” PhD thesis. L’institut National Polytechique de Lorrainne. see p. 83.

Dorato, P. C. Abdallah, and V. Cerone (1995). *Linear-Quadratic Control. An Introduction.* Prentice Hall, Englewood Cliffs, New Jersey 07632. see p. 104.

Hinze, M. (2004). “A Variational Discretization Concept in Control Constrained Optimization – The Linear-Quadratic Case.” in: see p. 92.

Hinze, M. and A. Kauffmann (1998). “Control Concepts for Parabolic Eqations with an Application to the Control of Fluid Flow.” Preprint Nr. 603/1998. see p. 96.

Hinze, M. and S. Volkwein (2004). “POD Surrogate Models for Nonlinear Dynamical Systems: Error Estimates and Suboptimal Control.” SFB609 Preprint 27-2004. see pp. 78, 83, 100.

— (2005). *Error Estimates for Abstract Linear-Quadratic Optimal Control Problems Using POD.* tech. rep. 2/2005. Institutes for Mathematics (Graz University of Technology) & Institutes for Mathematics and Scientific Computing (University of Graz). see pp. 61, 83, 86, 98, 99.

Kelley, C. T. (1999). *Iterative Methods for Optimization.* SIAM Philadelphia. see pp. 83, 93, 94.

Kunisch, K. and S. Volkwein (2006). *POD for Optimality Systems.* tech. rep. IMA10-06. University of Graz. see pp. 40, 101.

Kunisch, K. S. Volkwein, and L. Xie (2004). “HJB-POD-Based Feedback Design for the Optimal Control of Evolution Problems.” in: *SIAM Journal Applied Dynamical Systems* 3.4, pp. 701–722. see p. 102.

Lions, J.-L. (1971). *Optimal Control of Systems Governed by Partial Differential Equations.* Springer-Verlag, Berlin [u.a]. see pp. 83–86, 89, 91, 101, 103, 104.

Sachs, E. and W. A. Gruver (1980). *Algorithmic Methods in Optimal Control.* Pitman, Boston [u.a.] see p. 94.

Schütz, T. A. (2007). “Das Newton-Verfahren zur Lösung von Optimierungsproblemen mit parabolischen Differentialgleichungen.” MA thesis. Institut f. Numerische u. Angewandte Mathematik, University of Göttingen. see p. 93.

Troeltzsch, F. (2006). *Optimalsteuerung partieller Differentialgleichugen.* Vieweg. see pp. 83, 87–89, 91, 92, 94.

Volkwein, S. (2006). “Model Reduction Using POD.” Lecture Notes. see pp. 8, 35, 47, 48, 65, 71, 78, 86, 101, 103, 104, 137.

Chapter 7: Summary, Discussion and Outlook

Henri, T. (2003). "Utilisation de la Decomposition Orthogonale Propre pour le controle des problemes paraboliques." see pp. 137, 138.

Holmes, P. J. L. Lumley, and G. Berkooz (1996). *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press. see pp. 27, 32, 133, 139, 142, 144.

Ito, K. and S. Ravindran (1997). "A Reduced Basis Method for Control Problems Governed by PDEs." in: see p. 139.

Kahlbacher, M. (2006). "POD for Parameter Estimation of Bilinear Elliptic Problems." MA thesis. University of Graz. see pp. 47, 55, 137.

Kato, T. (1980). *Perturbation Theory for Linear Operators*. Springer-Verlag, Berlin [u.a]. see pp. 39, 44, 138.

Volkwein, S. (1999). "POD and SVD." see pp. 8, 47, 58, 80, 138, 139, 143.

— (2001*a*). "Optimal and Suboptimal Control of PDE: Augmented Lagrange-SQP Methods and Reduced-Order Modeling with POD." Habilitation Thesis. University of Graz. see p. 137.

— (2006). "Model Reduction Using POD." Lecture Notes. see pp. 8, 35, 47, 48, 65, 71, 78, 86, 101, 103, 104, 137.

Volkwein, S. and S. Weiland (2005). "An Algorithm for Galerkin Projections in Both Time and Spatial Coordinaties." see p. 139.

Chapter 8: Statistical Interpretation of the POD Method

Aubry, N. (1991). "On the Hidden Beauty of the POD." in: *Theoretical and Computational Fluid Dynamics* 2. Provided by the Smithsonian/NASA Astrophysics Data System, pp. 339–352. see p. 151.

Aubry, N. R. Guyonnet, and R. Lima (1991). "Spatiotemporal Analysis of Complex Signals: Theory and Applications." in: *Journal of Statistical Physics* 64.3-4, pp. 683–739. see pp. 146–148.

Holmes, P. J. L. Lumley, and G. Berkooz (1996). *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press. see pp. 27, 32, 133, 139, 142, 144.

Hussain, A. (1983). "Coherent Structures – Reality And Myth." in: *Physics of Fluids* 26.10, pp. 2816–2850. see p. 151.

Loève, M. (1955). *Probability Theory*. Van Nostrand. see p. 141.

Volkwein, S. (1999). "POD and SVD." see pp. 8, 47, 58, 80, 138, 139, 143.

— (2001*b*). "Optimal Control of a Phase-Field Model Using POD." in: *Zeitschrift für Angewandte Mathematik und Mechanik* 81.2, pp. 83–97. see pp. 27, 33, 35, 42, 142.