

FROM INTRUSIVE TO NON-INTRUSIVE MODEL
REDUCTION: STRUCTURE PRESERVATION AND EXACT
OPERATOR INFERENCE

HENRIK ROSENBERGER

September 2026

The research presented in this thesis was conducted at Centrum Wiskunde & Informatica (CWI) in Amsterdam. It is supported by the project “Discretize first, reduce next” (with project number VI.Vidi.193.105) of the research programme NWO Talent Programme Vidi.

TU/e



This document was typeset using the typographical look-and-feel *classicthesis* developed by André Miede and Ivo Pletikosić. The style was inspired by Robert Bringhurst’s seminal book on typography “*The Elements of Typographic Style*”. *classicthesis* is available for both L^AT_EX and L^yX:

<https://bitbucket.org/amiede/classicthesis/>

Cover: Wendy Bour - van Telgen

Image credits:

Wakes of wind turbines photograph by Christan Steiness, licensed under Creative Commons Attribution License.

A catalogue record is available from the Eindhoven University of Technology Library

ISBN: 978-90-386-6782-9

Printed by: Gildeprint

Henrik Rosenberger: *From intrusive to non-intrusive model reduction: Structure preservation and exact operator inference*, © September 2026

From intrusive to non-intrusive model reduction:
Structure preservation and exact operator inference

PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Technische
Universiteit Eindhoven, op gezag van de rector magnificus
prof.dr. S.K. Lenaerts, voor een commissie aangewezen door het
College voor Promoties, in het openbaar te verdedigen op
dinsdag 27 oktober 2026 om 13:30 uur

door

Henrik Karl Erich Rosenberger

geboren te Rottweil, Duitsland

Dit proefschrift is goedgekeurd door de promotoren en de samenstelling van de promotiecommissie is als volgt:

Voorzitter: prof.dr.ir. B.F. van Dongen
1e promotor: dr.ir. B. Sanderse
2e promotor: dr. G. Stabile (Sant'Anna School of Advanced Studies)
leden: dr. B. Peherstorfer (New York University)
prof.dr.rer.nat. D. Hartmann (Technische Universität Darmstadt)
prof.dr. K.P. Veroy- Grepl
prof.dr.ir. E.H. van Brummelen

Het onderzoek of ontwerp dat in dit proefschrift wordt beschreven is uitgevoerd in overeenstemming met de TU/e Gedragscode Wetenschapsbeoefening.

PREFACE

This thesis is the result of curiosity-driven research on projection-based model reduction. While research goals were defined and steps towards these goals were planned, the most exciting moments were those when new insights lead into alternative directions and, vice versa, pursued directions revealed unexpected insights.

Already at the start I had such an exciting moment. While I had planned to spend two weeks on preparing a chapter of my master thesis on structure-preserving ROMs for the incompressible Navier-Stokes equations [88] for publication, I discovered energy conservation as an additional preserved structure of the proposed ROMs. I investigated this property in more detail in the subsequent months and additionally discovered an equivalence between ROMs with and without pressure computation. These findings were combined in my first journal publication [90] presented in Chapter 2.

Next, I explored several ideas at the interface of ROMs and closure models for turbulence. After none of them proved useful, I went back to the structure-preserving ROM for the incompressible Navier-Stokes equations and aimed at reconstructing this ROM non-intrusively with operator inference. For this purpose, I studied among other things different definitions of the energy conservation property of convection operators. These investigations culminated in a famous anecdote within my group known as *the King's night*. At the eve of King's Day 2024 around 6, my colleagues asked me to join them for a beer. I replied that I would join them at the bar right after finishing some small proof step I was busy with. When I was done, my whiteboard was cluttered with sketches, it was 2 am and the bar had closed. Two days later, I assessed that there was no immediate use of the proof and put it aside.

Enforcing structure preservation for the non-intrusive ROM was rather straightforward and I was able to show preliminary results at the ECCOMAS 2024 conference where I got very valuable advice from the operator inference community. However, even with this advice, the accuracy of my numerical results remained unsatisfactory. I did discover that this inaccuracy was caused by the lack of information in the snapshot data, but I was not able to generate alternative snapshot data that does contain all required information. More precisely, I was not able to generate such alternative data along long trajectories like all works did that I had read. Generating such data via an ensemble of trajectories, in contrast, was completely obvious to me thanks to my King's night exercise.

However, I assumed that this ensemble approach was impractical. It took me several days to realize first that this approach actually can be used in practice and second that generating sufficiently informative snapshot data is not only challenging for me but for anyone who wants to apply operator inference to nonlinear problems. For this reason, I decided to put the project on structure-preserving operator inference for the incompressible Navier-Stokes equations on hold and to pursue the publication of the method for generating sufficiently informative snapshot data instead. While the method itself seemed obvious to me for quadratic problems, I spent several months on proving rigorously that the snapshot data generated by this method is indeed sufficiently informative for problems whose right-hand side f comprises an arbitrary set of monomials. It took me until the evening after the farewell symposium of my master thesis supervisor Barry Koren to find the last puzzle piece of this proof. I am convinced that it was the impressions of Barry's achievements that gave me the pivotal inspiration. The proof and the method itself are published [92] and presented in Chapter 3.

After submission of this manuscript, I explored whether the method can be extended to parametric problems. I found out swiftly that this extension is straightforward and not worth a publication on its own. However, I also discovered that this extension simplifies the parametric operator inference problem in a way that allows to decouple this problem. The article about both the extension and the decoupling is published [91] and presented in Chapter 4.

Besides these first-authored journal articles, I have published a conference proceedings article [89] and collaborated with two PhD students that performed internships at CWI with projects on model reduction in context of Rayleigh-Bénard convection [27] and adaptive time-stepping, [83], respectively.

CONTENTS

1	INTRODUCTION	1
1.1	Numerical simulation	1
1.2	Projection-based model reduction	4
1.3	Structure-preserving model reduction	6
1.4	Non-intrusive model reduction	7
1.5	Parametric model reduction	8
1.6	Thesis outline	9
2	ENERGY-CONSISTENT VELOCITY-ONLY ROMS FOR NAVIER-STOKES	11
2.1	Introduction	11
2.2	Preliminaries: continuous equation and FOM	14
2.2.1	The incompressible Navier-Stokes equations	14
2.2.2	Full order model: Finite volume discretization	15
2.3	ROM for time-dependent boundary conditions	17
2.3.1	Lifting function inspired by Helmholtz-Hodge decomposition	17
2.3.2	Velocity-only ROM	18
2.3.3	Efficient evaluation of the ODE right-hand side	19
2.3.4	Exact offline decomposition using matricized third-order tensors	21
2.3.5	Pressure recovery	22
2.4	Kinetic energy evolution	22
2.5	Equivalent velocity-pressure ROM	24
2.5.1	Velocity-pressure ROM	24
2.5.2	Equivalence to the proposed velocity-only ROM	25
2.5.3	Comparison with standard POD velocity-pressure ROMs	28
2.6	Numerical experiments	29
2.6.1	Setup	29
2.6.2	Results	32
2.7	Conclusion	37
3	EXACT OPERATOR INFERENCE WITH MINIMAL DATA	39
3.1	Introduction	39
3.2	Preliminaries	41
3.2.1	Dynamical systems with polynomial terms	42
3.2.2	Intrusive projection-based model reduction	42
3.2.3	Operator inference	43
3.2.4	Insufficiently informative snapshot data	45
3.3	Exact operator inference	46
3.3.1	Rank-ensuring ROM states and input signals	46

3.3.2	Snapshot data generation via single-step ensemble	48
3.3.3	Exact reconstruction of the intrusive ROM operators	49
3.3.4	Assumptions on the FOM solver	51
3.3.5	Remarks on changing ROM dimension	53
3.3.6	Choice of time step size Δt	53
3.4	Numerical experiments	54
3.4.1	Chafee-Infante equation / Allen-Cahn equation .	55
3.4.2	Ice sheet model	56
3.4.3	Burgers' equation	58
3.5	Conclusion	59
4	EXACTOPINF DECOUPLES PARAMETRIC PROBLEMS	61
4.1	Introduction	61
4.2	Preliminaries	63
4.2.1	Parametric dynamical system	63
4.2.2	Parametric Galerkin ROM	63
4.2.3	Parametric operator inference	65
4.2.4	Exact operator inference	66
4.3	Exact parametric operator inference	67
4.4	Decoupling of exact parametric operator inference . . .	70
4.5	Alternative derivation via Lagrange interpolation . . .	73
4.6	Numerical experiments	74
4.6.1	Heat equation with affine conductivity	75
4.6.2	Heat equation with non-affine conductivity . . .	76
4.7	Conclusion	79
5	CONCLUSIONS	83
5.1	Conclusions	83
5.2	Recommendations	84
5.3	Broader outlook	86
A	SUPPLEMENTARY MATERIAL FOR CHAPTER 2	87
A.1	Importance of orthogonal lifting function	87
A.2	Effect of energy-conserving Runge-Kutta methods . . .	88
A.3	Best approximation error relations	90
B	SUPPLEMENTARY MATERIAL FOR CHAPTER 3	93
B.1	Proof of full rank of operator inference matrix	93
B.1.1	Outline of proof	94
B.1.2	Multivariate polynomial interpolation without gaps	94
B.1.3	Homogeneous multivariate polynomial interpolation	95
B.1.4	Specific gappy multivariate polynomial interpolation	97
B.1.5	Gappy multivariate polynomial interpolation . .	100
B.2	Algorithm for generating ROM state set	101
B.3	ROM state errors with respect to the FOM	102

C	SUPPLEMENTARY MATERIAL FOR CHAPTER 4	105
C.1	Decoupling for arbitrary right-hand side	105
C.1.1	Parametric dynamical system and Galerkin ROM	105
C.1.2	Parametric operator inference	105
C.1.3	Decoupling parametric operator inference	106
C.2	ROM state errors with respect to the FOM	109
	BIBLIOGRAPHY	111
	LIST OF PUBLICATIONS	123
	CURRICULUM VITAE	125
	ACKNOWLEDGEMENTS	127
	SUMMARY	129

INTRODUCTION

1.1 NUMERICAL SIMULATION

Many processes relevant in science and engineering are described by partial differential equations. The wakes of wind turbines shown in Fig. 1.1, for example, can be described by the incompressible Navier-Stokes equations. These equations,

$$\nabla \cdot \mathbf{u} = 0, \quad (1.1)$$

$$\frac{\partial}{\partial t} \mathbf{u} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f} \quad (1.2)$$

describe the velocity $\mathbf{u}$ and pressure p of a fluid subject to an external force $\mathbf{f}$ in a domain Ω over a time interval $[0, T]$. For many of these partial differential equations, we do not know how to obtain an analytical solution in all but very simple settings. For the Navier-Stokes equations, we do not even know whether a solution exists [35].

Instead of analytical solutions, we therefore resort to numerical solutions of these partial differential equations to gain insights in the processes that they describe. For this purpose, the equations are discretized in space using methods such as the Finite Element methods



Figure 1.1: The incompressible Navier-Stokes equations can be used to describe the wakes of wind turbines (Courtesy: Vattenfall. Photographer is Christian Steiness). Reprinted from [54] under Creative Commons Attribution License.

[120] or the Finite Volume methods [33] and in time using methods such as Runge-Kutta methods or multi-step methods [53]. Concretely, a grid is defined on the domain Ω such that functions that describe velocity $\mathbf{u}$ and pressure p on the whole domain can be approximated by a function on this grid with only a finite number of degrees of freedom. As an example, Fig. 1.2 shows how velocity values u and v and pressure values p are located on a staggered grid for a finite-volume method [98]. Not only the functions, but also their spatial derivatives in (1.1) and (1.2) are approximated on this grid, resulting in the differential-algebraic equations

$$\mathbf{M}_h \mathbf{V}_h(t) = \mathbf{0}, \quad (1.3)$$

$$\Omega_h \frac{d}{dt} \mathbf{V}_h(t) = \mathbf{F}_h(\mathbf{V}_h(t), \mathbf{p}_h(t)). \quad (1.4)$$

with the vectors $\mathbf{V}_h$ and $\mathbf{p}_h$ that summarize velocity and pressure values, respectively. These equations can be solved numerically by integrating them in time starting from a given initial velocity $\mathbf{V}_h(0) = \mathbf{V}_h^0$ using a suitable time discretization method [97].

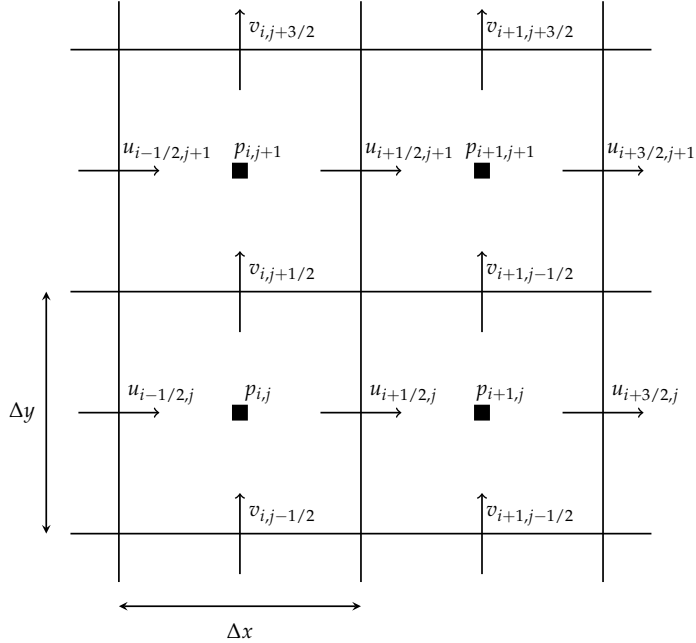


Figure 1.2: Staggered grid. Velocity values are located on the faces of the pressure finite volumes.

For problems of interest in science and engineering, the dimensions of the vectors $\mathbf{V}_h$ and $\mathbf{p}_h$ can be extremely large, so solving the equations

(1.3) and (1.4) can cause very large computational costs. Even if these costs are feasible for a single simulation, they can become prohibitive in tasks such as optimization, data assimilation, uncertainty quantification and the development of digital twins that require multiple simulations in similar settings. Several approaches have been developed to reduce the computational costs of such tasks by approximating high-fidelity models such as (1.3) and (1.4). In [63], these approaches are categorized into simplified models, statistical data-fit models and reduced models.

Simplified models are obtained by revising the derivation of the high-fidelity models and making less costly choices, for example, by linearizing nonlinear terms, simplifying physics, or using coarser discretization grids [63]. For the incompressible Navier-Stokes equations, the direct numerical simulation (DNS) in (1.3) and (1.4) could, for example, be replaced by Large Eddy Simulation (LES) or Reynolds-averaged Navier-Stokes (RANS). LES uses a coarser grid than DNS but is complemented with a closure model that accounts for the small scales not represented on this coarse grid [96]. RANS averages the Navier-Stokes equations over time but is complemented with a model for the effect of the time fluctuations on the time average [116]. The development of such models with sufficient accuracy is an ongoing challenge and requires in-depth knowledge of the underlying physics.

On the other hand, statistical data-fit models and reduced models are constructed from data via generic methods that do not require in-depth knowledge of the underlying physics. Statistical data-fit models are mappings composed of generic functions such as neural networks or Gaussian processes [117] whose parameters are fitted to data. The underlying physics enter the model generally only via this training data and possibly via weak constraints during training, for example in physics-informed neural networks [87].

In contrast, reduced models are closely based on the underlying physics as they approximate the high-dimensional high-fidelity model, typically by projecting this model onto a lower-dimensional subspace. As described in [64], most methods to construct reduced models, also called reduced-order models, can be distinguished into state-approximation methods and system-theoretic methods. System-theoretic methods approximate the mapping between inputs and outputs of dynamical systems that is crucial for control theory. On the other hand, state-approximation methods approximate the state itself of both static and dynamical systems. In this thesis, we focus on projection-based reduced order models for dynamical systems.

1.2 PROJECTION-BASED MODEL REDUCTION

Like classical numerical methods, projection-based model reduction derives discrete models from first principles. However, while classical numerical methods result in generic discretizations, projection-based model reduction leverages data to construct more efficient, problem-specific discretizations.

Consider the dynamical system

$$\dot{x}(t) = f(x(t)), \quad x(0) = x_0, \quad (1.5)$$

as it results from a classical spatial discretization of a partial differential equation, for example (1.4). We call this system the Full-Order Model (FOM). The dimension N of the state $x(t) \in \mathbb{R}^N$ can be very large, so integrating this FOM is very expensive. However, we can approximate this state within a linear subspace of much lower dimension n . Let $\mathbf{V}_n$ be a matrix whose columns span this linear subspace, then $\mathbf{V}_n \mathbf{V}_n^T x(t)$ defines the projection of $x(t)$ onto this subspace. For given state snapshots $x(t_1), \dots, x(t_K)$ at time instances $t_1, \dots, t_K$, Proper Orthogonal Decomposition (POD) computes the orthogonal matrix $\mathbf{V}_n$ that minimizes the error between these snapshots and their projection onto the linear subspace [57],

$$\sum_{i=1}^K \left\| x(t_i) - \mathbf{V}_n \mathbf{V}_n^T x(t_i) \right\|_2^2. \quad (1.6)$$

We call $\mathbf{V}_n$ the POD basis of dimension n . Within the linear subspace spanned by the columns of $\mathbf{V}_n$, the state $x(t)$ can be approximated as $x_r(t) = \mathbf{V}_n \tilde{x}(t)$ where $\tilde{x}(t)$ is defined via projection of the dynamical system (1.5),

$$\dot{\tilde{x}} = \mathbf{V}_n^T f(\mathbf{V}_n \tilde{x}(t)), \quad \tilde{x}(0) = \mathbf{V}_n^T x(0). \quad (1.7)$$

*Project FOM onto a
lower-dimensional
subspace to obtain
lower-dimensional
ROM*

We call this system a Reduced-Order Model (ROM). The dimension of this ROM is only n , so the computational costs of integrating this ROM are potentially much smaller than for the FOM. However, the construction of the ROM involves expensive computations such as the generation of the state snapshots $x(t_1), \dots, x(t_K)$, the computation of the POD basis $\mathbf{V}_n$ and the projection of the initial condition, $\mathbf{V}_n^T x(0)$. To benefit from the cost advantage of the ROM integration, we hence need to separate the expensive ROM construction costs from the integration. This separation is done by distinguishing two separate phases: an offline phase that comprises all expensive computations for the ROM construction, and an online phase that integrates the ROM efficiently. Because the expensive offline phase is only performed once, this setup

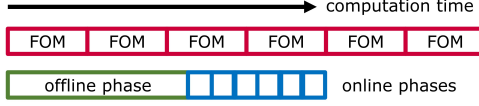


Figure 1.3: The ROM construction pays off after a few simulations.

pays off when the ROM is integrated multiple times as as illustrated in Fig. 1.3.

To integrate the ROM efficiently, we have to evaluate the right-hand side $\mathbf{V}_n^T f(\mathbf{V}_n \tilde{\mathbf{x}}(t))$ with a complexity independent of the FOM dimension N . If f is a linear function, so $f(\mathbf{x}) = \mathbf{A}\mathbf{x}$ for some matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$ then the ROM right-hand side

$$\mathbf{V}_n^T f(\mathbf{V}_n \tilde{\mathbf{x}}) = \mathbf{V}_n^T \mathbf{A} \mathbf{V}_n \tilde{\mathbf{x}}$$

can be evaluated efficiently as $\tilde{\mathbf{A}} \tilde{\mathbf{x}}$ with the matrix

$$\tilde{\mathbf{A}} = \mathbf{V}_n^T \mathbf{A} \mathbf{V}_n \in \mathbb{R}^{n \times n}.$$

The construction of this matrix $\tilde{\mathbf{A}}$ is expensive but can be computed during the offline phase. Similarly, if the FOM right-hand side is a quadratic function $f(\mathbf{x}) = \mathbf{H}(\mathbf{x} \otimes \mathbf{x})$ with the matrix $\mathbf{H} \in \mathbb{R}^{N \times N^2}$ then the ROM right-hand side

$$\mathbf{V}_n^T f(\mathbf{V}_n \tilde{\mathbf{x}}) = \mathbf{V}_n^T \mathbf{H}((\mathbf{V}_n \tilde{\mathbf{x}}) \otimes (\mathbf{V}_n \tilde{\mathbf{x}}))$$

can be evaluated as $\tilde{\mathbf{H}}(\tilde{\mathbf{x}} \otimes \tilde{\mathbf{x}})$ with the offline-computed matrix

$$\tilde{\mathbf{H}} = \mathbf{V}_n^T \mathbf{H}(\mathbf{V}_n \otimes \mathbf{V}_n) \in \mathbb{R}^{n \times n^2}.$$

Note that the matrices $\mathbf{A}$ and $\mathbf{H}$ are typically sparse, so the corresponding right-hand sides f can be evaluated in $\mathcal{O}(N)$. In contrast, the ROM matrices $\tilde{\mathbf{A}}$ and $\tilde{\mathbf{H}}$ are in general dense, so the computational costs of evaluating the ROM right-hand sides $\tilde{\mathbf{A}} \tilde{\mathbf{x}}$ and $\tilde{\mathbf{H}}(\tilde{\mathbf{x}} \otimes \tilde{\mathbf{x}})$ are $\mathcal{O}(n^2)$ and $\mathcal{O}(n^3)$, respectively. Similarly, an l -th order polynomial ROM right-hand side can be evaluated with complexity $\mathcal{O}(n^{l+1})$. Consequently, these costs can be prohibitive for sufficiently large n and l , despite being independent of the FOM dimension N . In such cases, and for non-polynomial FOM right-hand sides f , hyper-reduction techniques introduce an approximation of the ROM right-hand side $\mathbf{V}_n^T f(\mathbf{V}_n \tilde{\mathbf{x}}(t))$ that can be evaluated efficiently [24, 29, 32].

The described projection-based model reduction approximates the FOM in two ways. First, the state space is restricted to the linear subspace spanned by the columns of the ROM basis. Second, the FOM

dynamics are approximated by projecting the FOM right-hand side onto the same linear subspace. While the POD basis minimizes the error introduced by the first approximation,

$$\|x(t) - \mathbf{V}_n \mathbf{V}_n^T x(t)\|_2, \quad (1.8)$$

the error introduced by the second approximation,

$$\|f(x(t)) - \mathbf{V}_n \mathbf{V}_n^T f(\mathbf{V}_n \mathbf{V}_n^T x(t))\|_2, \quad (1.9)$$

is not taken into account in the definition of this basis. As a consequence, the second error can be substantial even when the first error is very small. For example, small scales play an important role in turbulent flows. These scales are neglected by the POD basis since they carry only little energy and states can quite accurately be represented without these scales, but the projected dynamics lack the energy dissipation of these small scales and tend to be unstable [37]. Since POD does not take the physics described by the FOM dynamics into account and neglects the small scales, the projection fails to capture these small-scale physics well.

1.3 STRUCTURE-PRESERVING MODEL REDUCTION

*Enforce physical
structure in the ROM
construction*

An attempt to compensate for this weak spot of POD-based ROMs is the concept of structure preservation. While the physics of the system are not taken into account by the POD and only approximated in the projection, physical properties are explicitly incorporated in the construction of the ROM and hence satisfied exactly. Examples for such physical structures are Lagrangian, Hamiltonian and port-Hamiltonian structure [64]. In the incompressible Navier-Stokes equations, kinetic energy conservation is such a physical structure. For homogeneous boundary conditions, without diffusion and without forcing, the kinetic energy of the continuous equations (1.1) and (1.2) is conserved. For suitable discretizations, this energy conservation has been preserved in the FOM (1.3) and (1.4) [112] and also in the ROM [98]. As a result, this ROM is better at representing the behavior qualitatively and guaranteed stable.

However, the ROM in [98] is limited to constant boundary conditions and the described structure preservation holds only for homogeneous boundary conditions. Boundary conditions can add or remove kinetic energy to the system, so enforcing that the ROM energy does not increase would be unphysical. The first research goal in this thesis is therefore:

1. **A structure-preserving ROM for the incompressible Navier-Stokes equations with time-dependent boundary conditions.**

To achieve this goal, Chapter 2 generalizes the concept of energy conservation to the concept of energy consistency: due to boundary conditions, the kinetic energy does not remain constant, but the kinetic energy evolution of the proposed structure-preserving ROM has the same structure as the kinetic energy evolution of the FOM. For this purpose, a lifting function proposed in [98] for constant boundary conditions is generalized to time-dependent boundary conditions. This lifting function is orthogonal to the homogeneous velocity that remains after subtracting the lifting function from the velocity. Preserving this orthogonality and the orthogonality between the homogeneous velocity and the pressure gradient in the ROM leads to energy consistency.

1.4 NON-INTRUSIVE MODEL REDUCTION

While projection-based model reduction has been investigated for decades in academia [85, 111], it is only rarely adopted in industry. The main reason for this discrepancy lies in the different software used. While academics are free to individually choose the software for demonstrating their work or can even write the code for their toy problems themselves, engineers in industry are typically constrained to use software chosen to fit the needs of large groups of people. Such software is often much more sophisticated than academic software, but as a consequence also much less flexible and transparent. This lack of flexibility and transparency is an issue in the construction of projection-based ROMs. As described in Section 1.2, the evaluation of the ROM right-hand side $\mathbf{V}_n^T \mathbf{f}(\mathbf{V}_n \tilde{\mathbf{x}}(t))$ requires access to the FOM operator $\mathbf{f}$. For example, if $\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x}$, we need to compute the ROM matrix $\tilde{\mathbf{A}} = \mathbf{V}_n^T \mathbf{A} \mathbf{V}_n$. This computation is trivially possible, if we can access the FOM matrix $\mathbf{A}$. If not, we can compute $\tilde{\mathbf{A}}$ entry-wise, by evaluating the FOM right-hand side $\mathbf{f}$ for the j -th column of the POD basis $\mathbf{V}_n$, $\mathbf{v}_j$ and projecting it onto the i -th column $\mathbf{v}_i$, $\mathbf{v}_i^T \mathbf{f}(\mathbf{v}_j) = \mathbf{v}_i^T \mathbf{A} \mathbf{v}_j = \tilde{\mathbf{A}}_{ij}$. However, in industry software, neither access to $\mathbf{A}$ nor to $\mathbf{f}$ might be possible. To address this issue and enable adoption of projection-based model reduction in industry, non-intrusive model reduction investigates how ROM operators such as $\tilde{\mathbf{A}}$ can be computed without intrusive access to the FOM solver.

Operator inference computes such ROM operators non-intrusively for problems with polynomial right-hand side $\mathbf{f}$ [78]. Whether $\mathbf{f}$ is polynomial and even which polynomial degrees are involved can often be inferred from the underlying partial differential equation and hence can be determined without intrusive access to the FOM solver. This knowledge about the polynomial structure of $\mathbf{f}$ is then leveraged to fit ROM operators with the same polynomial structure to snapshot data. Such

*Operator inference:
construct polynomial
ROM operators without
intrusive FOM access*

snapshot data can be obtained non-intrusively from FOM simulations or from measurements of experiments and is also used to compute the POD basis. However, such snapshots are in general not fully consistent with the intrusive ROM and hence result in inferred ROM operators that only approximate the intrusive ROM operators. To reconstruct the intrusive ROM operators exactly, a method has been proposed that generates snapshots that are fully consistent with the intrusive ROM [77]. This method consequently enables exact reconstruction, but only if the generated snapshots are sufficiently informative. Concretely, the operator inference least squares matrix derived from these snapshots must have full rank. Existing works have achieved this full rank using heuristic strategies to generate snapshot data and checked for full rank a-posteriori, but without a guarantee of success. The second research goal of this thesis is therefore:

2. A non-intrusive method that is guaranteed to reconstruct intrusive ROMs exactly.

To achieve this goal, Chapter 3 presents a novel snapshot data generation method that first identifies ROM states that imply full rank and then generates the corresponding snapshots via an ensemble of simulations that are run for only a single time step. The resulting least squares matrix has full rank, so operator inference reconstructs the intrusive ROM operators exactly up to machine precision.

1.5 PARAMETRIC MODEL REDUCTION

In the previous three sections, we have only considered the use of a ROM to reproduce the same trajectory that has already been computed by the FOM to generate snapshot for the POD computation. Such a mere reproduction is of little practical use. The actual benefit of ROMs is to perform computations that have not been performed by the FOM. One example is the extrapolation of the ROM state $\tilde{x}(t)$ beyond the time instances $t_1, \dots, t_K$ for which the FOM computed snapshots. To illustrate other use cases, we extend the dynamical system (1.5) by a parameter μ ,

$$\dot{x}(t; \mu) = f(x(t; \mu); \mu), \quad x(0; \mu) = x_0(\mu). \quad (1.10)$$

For this system, we can generate snapshot data for a set $\{\mu_1, \dots, \mu_s\}$ of samples of parameter μ , construct a ROM from these snapshot data and evaluate this ROM for other values of μ . There are several approaches how to construct ROMs from these snapshot data and how to evaluate these ROMs for unseen values of μ [12]. One approach computes a single global basis $\mathbf{V}_n$ from all snapshots of all parameter samples and

assumes that the parameter-dependent right-hand side $f(x; \mu)$ can be decomposed as sum of parameter-independent operators $f^{(i)}(x)$ with parameter-dependent weights $\theta^{(i)}(\mu)$,

$$f(x; \mu) = \sum_{i=1}^q \theta^{(i)}(\mu) f^{(i)}(x). \quad (1.11)$$

For each operator $f^{(i)}$, the corresponding ROM operator $f_r^{(i)}$ can then be computed as described in Section 1.2. As described in the previous section, this computation is intrusive, so a non-intrusive alternative computation might be needed in practice. Operator inference can be generalized to parametric problems to serve as such an alternative if f is polynomial [70, 119]. However, as in the non-parametric case, the exact reconstruction of the intrusive ROM operators is not guaranteed for arbitrary snapshot data. As an additional issue, operator inference can in general not compute the individual operators $f_r^{(i)}$ separately as in the intrusive approach, but does compute them all at once. Such a monolithic computation is more expensive and can even dominate the offline ROM construction costs for sufficiently large ROM dimension n , polynomial degrees present in f and number q of terms in the sum (1.11). The third research goal of this thesis is therefore:

3. **A non-intrusive method that is guaranteed to reconstruct intrusive parametric ROMs with affine parameter dependency exactly with feasible computational costs.**

To achieve this goal, Chapter 4 extends the snapshot generation method in Chapter 3 to the parametric case. This extension guarantees the exact reconstruction of the intrusive ROM operators and enables to decouple the monolithic problem of inferring the ROM operators $f_r^{(i)}$ at once into smaller subproblems of inferring each $f_r^{(i)}$ separately and hence more efficiently.

1.6 THESIS OUTLINE

This thesis is structured as follows. Chapters 2, 3 and 4 are self-contained and present the articles [90], [92] and [91], respectively, in adapted form.

Chapter 2 presents a ROM for the incompressible Navier-Stokes equations with time-dependent boundary conditions that preserves the structure of the kinetic energy evolution. This ROM can simulate the velocity without computing the pressure but is equivalent to a ROM with pressure computation.

Chapter 3 presents a method to generate snapshot data for operator inference that guarantees the exact reconstruction of the corresponding

intrusive ROM operators. The amount of snapshot data generated by this method is minimal and simplifies the operator inference least squares problem to a linear system.

Chapter 4 presents the extension of this snapshot data generation method to parametric problems. Besides guaranteeing exact reconstruction of intrusive ROM operators, the method enables a decoupling of the parametric operator inference problems into smaller subproblems that can be solved more efficiently.

Conclusion are drawn in Chapter 5.

NO PRESSURE? ENERGY-CONSISTENT ROMS FOR THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS WITH TIME-DEPENDENT BOUNDARY CONDITIONS

This chapter presents an advance in the field of structure-preserving model reduction. An existing energy-conserving ROM for the incompressible Navier-Stokes equations is extended to time-dependent inhomogeneous boundary conditions. The resulting ROM preserves the structure of the kinetic energy evolution.

To achieve this structure preservation, the velocity is decomposed into two components: a velocity that satisfies homogeneous boundary conditions and a lifting function that satisfies the prescribed inhomogeneous boundary conditions. These two components are orthogonal to each other which is crucial to preserve the structure of the kinetic energy evolution.

The velocity decomposition also enables to compute the velocity without computing the pressure. The equivalence of this velocity-only ROM to a velocity-pressure ROM is shown.

Numerical experiments on test cases with inflow-outflow boundary conditions confirm the correctness and efficiency of the new ROM, and the equivalence with the velocity-pressure formulation.

2.1 INTRODUCTION

Turbulent flow is an ubiquitous phenomenon in fluid flows such as blood flow in arteries, ocean currents and air flow around wind turbines. Accurate numerical simulation of these turbulent flows requires computationally expensive methods such as Direct Numerical Simulation (DNS) [72] or Large Eddy Simulation (LES) [81]. The computational cost of these methods can sometimes be tolerated for a small number of simulations but becomes prohibitive when it comes to tasks that require many simulation runs such as in the case of optimization and uncertainty quantification studies.

To mitigate the computational costs, several techniques have been developed to reduce the complexity of the respective full order model (FOM) resulting in so-called reduced-order models (ROMs). Many of these techniques are projection-based, i.e., they project the high-dimensional discretization of the model obtained from classical discretization methods such as Finite Elements, Finite Volumes or Finite Differences onto a low-dimensional space, the reduced basis space.

This chapter is based on the following article:

Henrik Rosenberger and Benjamin Sanderse. "No pressure?"

Energy-consistent ROMs for the incompressible Navier-Stokes equations with time-dependent boundary conditions." In: Journal of Computational Physics 491 (2023), p. 112405. doi: 10.1016/j.jcp.2023.112405.

CRedit author statement

Henrik Rosenberger:
Conceptualization,
Methodology, Software,
Writing – original draft.
Benjamin Sanderse:
Conceptualization,
Funding acquisition,
Methodology, Software,
Writing – review & editing.

Methods to obtain such reduced bases include sampling strategies [51] and Proper Orthogonal Decomposition (POD) [114].

The POD, which is also known as Karhunen-Loève expansion and Principal Component Analysis, is optimal in the sense that it extracts the reduced basis from given high-resolution data which minimizes the approximation error with respect to this data in the l^2 norm [114]. However, several authors observed that ROMs resulting from the Galerkin projection of such POD reduced bases tend to exhibit instabilities when applied to fluid flow problems [37, 105]. This observation is often related to energy dissipation at the smallest scales of turbulent flow, which are not included in the POD reduced basis [37]. Hence, approaches to address the ROM instability include methods that add such small scale effects either by introducing additional dissipation via closure modeling [3, 115] or by using a POD based on a discrete equivalent of the H^1 norm instead of the l^2 norm in order to achieve a better representation of small scales in the reduced basis [58]. Alternative approaches use more dissipative ROM bases whose modes are weighted sums of the standard POD modes and randomly chosen higher-order POD modes, which are assumed to represent the small scales [7, 8].

Another cause of instability has been identified in the ROM pressure field. This pressure field can violate an inf-sup condition at the ROM level, even if the inf-sup condition of the underlying FOM is satisfied [9, 105]. When violating the condition, the ROM can exhibit spurious pressure modes that induce instability. Hence, many works have been developed to satisfy this condition on the ROM level [9, 22, 42, 94, 105]. At the same time, it has been shown that even for a linear ordinary differential equation (ODE), a stable FOM can result in an unstable POD-Galerkin ROM [84]. Consequently, POD-Galerkin ROM instabilities are not exclusively related to the incompressible Navier-Stokes equations.

A promising approach to avoid ROM instabilities, not specific to the incompressible Navier-Stokes equations, is the idea of preserving certain structure of the FOM such that stability is implied. A method widely used for compressible flows [10, 76, 93, 102] preserves the stability of an equilibrium point at the origin by performing POD with respect to energy-based inner products. For Hamiltonian systems, Proper Symplectic Decomposition was proposed as a modification of POD to obtain Galerkin ROMs that preserve symplecticity, so the Hamiltonian is time-invariant and the ROM is stable [79]. Alternative symplecticity-preserving bases have been proposed that can be computed more efficiently [1] and recover the FOM Hamiltonian exactly [44]. The concept of symplecticity preservation has been extended to model reduction on nonlinear manifolds [19] and non-canonical Hamiltonian systems [104]. Furthermore, non-intrusive methods have been developed for canonical and non-canonical Hamiltonian systems that

preserve the Hamiltonian structure [49]. For mechanical systems, Lagrangian structure-preserving methods have been proposed [25, 66] and observed to be stable [25]. Stability-preserving methods have also been proposed for port-Hamiltonian systems [28, 68] and second-order index-1 descriptor systems [13]. For metriplectic systems, a method has been proposed that preserves the metriplectic structure and is observed to be long-term stable [48]. For nonlinear conservation laws, stable ROMs have been proposed that preserve a semi-discrete entropy dissipation law [26].

For the incompressible Navier-Stokes equations, the POD-Galerkin ROM proposed in [98] achieves nonlinear stability by preserving the conservation of kinetic energy in the inviscid limit for homogeneous boundary conditions. In this work, this ROM is generalized to arbitrary time-dependent inhomogeneous boundary conditions, while possessing three properties: (i) structure preservation (consistency with the kinetic energy equation), (ii) velocity-only formulation (pressure-free), and (iii) arbitrary inhomogeneous boundary conditions (not separable in space- and time-dependent functions). To our best knowledge such a ROM formulation does not yet exist.

To achieve the first two properties, we propose the use of a specific lifting function. Classically, lifting functions are chosen as a velocity field that satisfies the prescribed inhomogeneous boundary conditions, so that the ROM can be solved with homogeneous boundary conditions [36, 46, 50, 71]. Inspired by the Helmholtz-Hodge decomposition [14], we propose a lifting function that is orthogonal to the homogeneous velocity field. This orthogonality allows us to preserve the structure of the kinetic energy equation on the ROM level.

In addition, this choice of lifting function has the benefit that the pressure is eliminated from the formulation, leading to a velocity-only ROM, i.e., the ROM velocity can be simulated without requiring the computation of the pressure. An important benefit of eliminating the pressure is that our velocity-only ROM is not affected by the inf-sup instability mentioned above. We stress that the omission of the pressure is *not* an approximation as performed in some early works [59, 80]. As shown in [74], merely neglecting the pressure in ROMs can lead to significant errors. In our ROM, in contrast, the elimination of the pressure is rigorous and exact. In fact, we show that the proposed velocity-only ROM is equivalent to a velocity-pressure ROM (for a specific choice of basis). This equivalence gives us valuable insights in the behaviour of the ROMs. Compared to other formulations that eliminate the pressure such as the streamfunction-vorticity formulation [43], our approach has the advantage of partial decoupling: while streamfunction and vorticity depend on each other, our lifting function does not depend on the remaining velocity component, but only on the prescribed boundary

conditions. This partial decoupling has many advantages, for example, time integration errors do not accumulate in the lifting function.

Lastly, to be able to deal with arbitrary inhomogeneous time-dependent boundary conditions, we propose to approximate the boundary conditions by projection onto a POD basis. This explicit approximation of boundary conditions is necessary because standard hyper-reduction of the ROM ODE right-hand side [17, 29] would not effect a sufficient efficiency improvement in context of our proposed lifting function.

This chapter is organized as follows. In Section 2.2, we introduce the continuous formulation and our finite volume discretization of the incompressible Navier-Stokes equations. In Section 2.3, we present our novel velocity-only ROM for time-dependent boundary condition. In Section 2.4, we show that this ROM preserves the structure of the kinetic energy evolution. In Section 2.5, we show the ROM's equivalence to a velocity-pressure ROM. In Section 2.6, we test the novel ROM in two test cases involving time-dependent inflow boundary conditions.

2.2 PRELIMINARIES: CONTINUOUS EQUATIONS AND FULL-ORDER MODEL

2.2.1 The incompressible Navier-Stokes equations

The incompressible Navier-Stokes equations describe the conservation of mass and momentum of a fluid in a domain $\Omega \subset \mathbb{R}^d$ over a time interval $[0, T]$, and can be expressed as

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega \times [0, T], \quad (2.1)$$

$$\frac{\partial}{\partial t} \mathbf{u} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = -\frac{1}{\rho} \nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f} \quad \text{in } \Omega \times [0, T], \quad (2.2)$$

with divergence-free initial condition

$$\mathbf{u}(x, 0) = \mathbf{u}_0(x) \quad \text{in } \Omega, \quad (2.3)$$

where the vector field $\mathbf{u} : \Omega \times [0, T] \rightarrow \mathbb{R}^d$ describes the velocity, the scalar field $p : \Omega \times [0, T] \rightarrow \mathbb{R}$ the pressure, ν is the kinematic viscosity, ρ the (constant) density (which will be absorbed in the pressure in subsequent equations), and $\mathbf{f}$ a forcing term (possibly space- and time-dependent). A set of boundary conditions which is appropriate to ensure uniqueness of the solution $\mathbf{u}$ (see [47, Section 3.8]) is assumed to be prescribed.

2.2.2 Full order model: Finite volume discretization

The Navier-Stokes equations (2.1)-(2.2) in $d = 2$ dimensions are spatially discretized with the second-order finite volume method on a staggered grid proposed in [98] and depicted in Fig. 2.1. The extension to $d = 3$ is straightforward, and all the methods described in the subsequent sections also apply to that case.

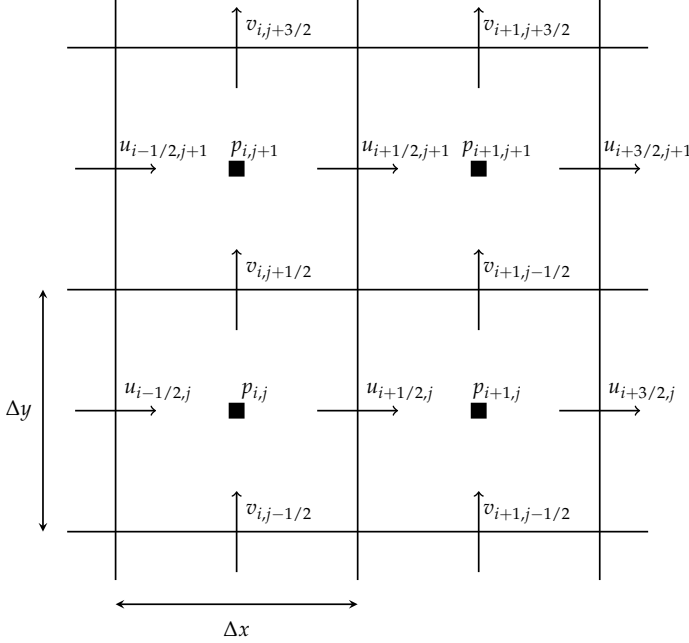


Figure 2.1: Staggered grid. Velocity values are located on the faces of the pressure finite volumes.

The u - and v -component of velocity, and the pressure p are discretized by $\mathbf{u}_h(t) \in \mathbb{R}^{N_u}$, $\mathbf{v}_h(t) \in \mathbb{R}^{N_v}$ and $\mathbf{p}_h(t) \in \mathbb{R}^{N_p}$, respectively, where $N_p = N_x \times N_y$ is the number of “pressure finite volumes” and $N_u, N_v \approx N_p$ are the numbers of “velocity component finite volumes” which depend on the type of boundary conditions. The velocity component vectors are aggregated in one velocity vector $\mathbf{V}_h(t) \in \mathbb{R}^{N_V}$, $N_V = N_u + N_v$. The spatial discretization of (2.1)-(2.2) can be summarized as

$$\mathbf{M}_h \mathbf{V}_h(t) = \mathbf{y}_M(t) := \mathbf{F}_M \mathbf{y}_{bc}(t), \quad (2.4)$$

$$\mathbf{\Omega}_h \frac{d}{dt} \mathbf{V}_h(t) = \mathbf{F}_h^{CD}(\mathbf{V}_h(t), \mathbf{y}_{bc}(t)) - \mathbf{G}_h \mathbf{p}_h(t). \quad (2.5)$$

Here, $\mathbf{\Omega}_h \in \mathbb{R}^{N_V \times N_V}$ is a diagonal matrix with the sizes of the velocity component finite volumes corresponding to the entries of $\mathbf{V}_h$ on its

diagonal. The matrix $\mathbf{M}_h \in \mathbb{R}^{N_p \times N_V}$ describes the mass equation (2.1) such that each row of (2.4) represents conservation of mass over the finite volume around a pressure point $p_{i,j}$ as depicted in Figure 2.1,

$$u_{i+1/2,j}\Delta y - u_{i-1/2,j}\Delta y + v_{i,j+1/2}\Delta x - v_{i,j-1/2}\Delta x = 0. \quad (2.6)$$

using the midpoint rule.

The vector $\mathbf{y}_{bc}(t) \in \mathbb{R}^{N_{bc}}$ constitutes the time-dependent parametrization of inhomogeneous boundary conditions. In this chapter, we only consider time-dependence of inflow boundary conditions. Hence, $\mathbf{y}_{bc}(t)$ comprises the prescribed Dirichlet velocity values at finite volume faces on inflow boundaries and N_{bc} is the number of such faces. However, the construction of $\mathbf{y}_{bc}(t)$ can be generalized to test cases with other types of time-dependent boundary conditions. Note that N_{bc} scales only by $\mathcal{O}(h^{-d+1})$, while N_V and N_p scale by $\mathcal{O}(h^{-d})$ upon decreasing the mesh size h .

The matrix $\mathbf{F}_M \in \mathbb{R}^{N_p \times N_{bc}}$ maps the vector $\mathbf{y}_{bc}(t)$ to the mass equation right-hand side $\mathbf{y}_M(t) \in \mathbb{R}^{N_p}$. The term $\mathbf{F}_h^{CD}(\mathbf{V}_h(t), \mathbf{y}_{bc}(t))$ contains the discretized convection, diffusion and forcing terms as well as all boundary condition contributions. The discretized forcing term $\mathbf{f}(t) \in \mathbb{R}^{N_V}$ is assumed fixed and time-independent, hence neither $\mathbf{f}(t)$ nor t appear as an argument of $\mathbf{F}_h^{CD}$. The term $\mathbf{G}_h \mathbf{p}_h$ is the discretization of the (integrated) pressure gradient ∇p in (2.2) and $\mathbf{G}_h \in \mathbb{R}^{N_V \times N_p}$ satisfies the discrete compatibility relation between the divergence and gradient operator

$$\mathbf{G}_h = -\mathbf{M}_h^T, \quad (2.7)$$

for all types of boundary conditions. This is a crucial property of the staggered grid discretization that will be used at several points in the construction of our new energy-consistent ROM approach for time-dependent boundary conditions.

To solve the system (2.4)-(2.5), we apply the discrete divergence operator $\mathbf{M}_h$ to the Ω_h^{-1} -premultiplied momentum equation (2.5) and insert the mass equation (2.4) to get the Poisson equation for the pressure,

$$\begin{aligned} \mathbf{L}_h \mathbf{p}_h(t) &:= \mathbf{M}_h \Omega_h^{-1} \mathbf{G}_h \mathbf{p}_h(t) \\ &= -\frac{d}{dt} \mathbf{y}_M(t) + \mathbf{M}_h \Omega_h^{-1} \mathbf{F}_h^{CD}(\mathbf{V}_h(t), \mathbf{y}_{bc}(t)). \end{aligned} \quad (2.8)$$

As a result, when integrating the velocity $\mathbf{V}_h(t)$ according to the momentum equation (2.5), the pressure $\mathbf{p}_h(t)$ computed from (2.8) ensures the velocity to satisfy the mass equation (2.4).

Note that this approach requires the invertibility of $\mathbf{L}_h$. As described in [99], $\mathbf{L}_h$ is singular for certain boundary conditions and the pressure

p_h is only defined up to an additional constant. However, by adding a constraint, we can construct an invertible matrix $\tilde{\mathbf{L}}_h \in \mathbb{R}^{N_p \times N_p}$ such that the unique solution to the regularized Poisson equation

$$\tilde{\mathbf{L}}_h p_h = -\frac{d}{dt} \mathbf{y}_M(t) + \mathbf{M}_h \mathbf{\Omega}_h^{-1} \mathbf{F}_h^{CD}(\mathbf{V}_h(t), \mathbf{y}_{bc}(t)) \quad (2.9)$$

is also a solution to the Poisson equation (2.8). For boundary conditions for which $\mathbf{L}_h$ is singular, we therefore compute p_h via (2.9) to simulate the FOM (2.4)-(2.5).

To avoid the need to distinguish between cases with $\mathbf{L}_h$ singular and invertible in our notation, we consider (2.9) the equation to be solved within the FOM simulation, where we set $\tilde{\mathbf{L}}_h := \mathbf{L}_h$ for boundary conditions for which $\mathbf{L}_h$ is invertible. In both cases, the inverse of $\tilde{\mathbf{L}}_h$ satisfies

$$\mathbf{L}_h \tilde{\mathbf{L}}_h^{-1} \mathbf{y} = \mathbf{y} \quad \text{for all } \mathbf{y} \in \text{Im}(\mathbf{L}_h), \quad (2.10)$$

where $\text{Im}(\mathbf{F}_M) \subset \text{Im}(\mathbf{M}_h) = \text{Im}(\mathbf{L}_h)$ and

$$\mathbf{y}^T \tilde{\mathbf{L}}_h^{-T} \mathbf{z} = \mathbf{y}^T \tilde{\mathbf{L}}_h^{-1} \mathbf{z}. \quad (2.11)$$

In other words, within the image space of $\mathbf{M}_h$, $\tilde{\mathbf{L}}_h^{-1}$ is symmetric and acts as an inverse to $\mathbf{L}_h$.

Suitable time discretizations of the system consisting of (2.5) and (2.9) are, for example, described in [97, 99].

2.3 NOVEL VELOCITY-ONLY AND ENERGY-CONSISTENT ROM FOR TIME-DEPENDENT BOUNDARY CONDITIONS

2.3.1 Lifting function inspired by Helmholtz-Hodge decomposition

To construct a ROM for the FOM described in Section 2.2.2, we generalize the ideas in [98, Section 4] to time-dependent boundary conditions. We decompose the FOM velocity,

$$\mathbf{V}_h(t) = \mathbf{V}_{h,\text{hom}}(t) + \mathbf{V}_{\text{inhom}}(t), \quad (2.12)$$

such that the time-dependent lifting function $\mathbf{V}_{\text{inhom}}(t)$ satisfies the mass equation,

$$\mathbf{M}_h \mathbf{V}_{\text{inhom}}(t) = \mathbf{y}_M(t), \quad (2.13)$$

and consequently, the homogeneous component $\mathbf{V}_{h,\text{hom}}(t)$ satisfies the homogeneous mass equation,

$$\mathbf{M}_h \mathbf{V}_{h,\text{hom}}(t) = \mathbf{0}. \quad (2.14)$$

The choice of the lifting function $V_{\text{inhom}}(t)$ is of pivotal importance. Inspired by the Helmholtz-Hodge decomposition [14], we choose the lifting function as

$$V_{\text{inhom}}(t) = \Omega_h^{-1} \mathbf{G}_h \bar{\mathbf{L}}_h^{-1} \mathbf{y}_M(t) = \Omega_h^{-1} \mathbf{G}_h \bar{\mathbf{L}}_h^{-1} \mathbf{F}_M \mathbf{y}_{\text{bc}}(t). \quad (2.15)$$

This choice has three decisive properties. First, the lifting function indeed satisfies the mass equation with the inhomogeneous right-hand side, (2.13). This property implies the homogeneous mass equation (2.14) for the homogeneous component $V_{h,\text{hom}}(t)$. We use this homogeneous mass equation to construct a velocity-only ROM in the next section.

FOM: Lifting function
 (1) takes care of
 boundary conditions,
 (2) can be computed
 efficiently and (3) is
 orthogonal to
 homogeneous velocity

Second, the lifting function is a linear function of the boundary condition vector $\mathbf{y}_{\text{bc}}(t)$. We use this property to compute the ROM efficiently for arbitrary time-dependent boundary conditions in Section 2.3.3.

Third, as in the Helmholtz-Hodge decomposition, the components in (2.12) are orthogonal, namely with respect to the Ω_h -inner product,

$$\begin{aligned} V_{h,\text{hom}}^T \Omega_h V_{\text{inhom}} &= V_{h,\text{hom}}^T \Omega_h \Omega_h^{-1} \mathbf{G}_h \bar{\mathbf{L}}_h^{-1} \mathbf{y}_M(t) \\ &= (\mathbf{M}_h V_{h,\text{hom}})^T \bar{\mathbf{L}}_h^{-1} \mathbf{y}_M(t) = 0. \end{aligned} \quad (2.16)$$

This property guarantees the energy consistency of the ROM as described in Section 2.4.

2.3.2 Velocity-only ROM

To approximate the homogeneous velocity $V_{h,\text{hom}}(t)$, we construct a POD basis with respect to the Ω_h -inner product computed from snapshots $V_{h,\text{hom}}^j = V_h^j - V_{\text{inhom}}(t^j)$, where V_h^j is the time-discrete approximation of the velocity $V_h(t)$ defined by the FOM (2.4)-(2.5) at time step t^j . Due to the homogeneous mass equation (2.14), the resulting Ω_h -orthonormal POD basis $\Phi_{\text{hom}} \in \mathbb{R}^{N_V \times R_{\text{hom}}}$ satisfies

$$\mathbf{M}_h \Phi_{\text{hom}} = \mathbf{0}, \quad (2.17)$$

as described in [98]. Hence, we can construct the low-dimensional approximation $V_{r,\text{hom}}(t) = \Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) \approx V_{h,\text{hom}}(t)$ such that the ROM $V_r(t) = V_{r,\text{hom}}(t) + V_{\text{inhom}}(t)$ satisfies the mass equation,

$$\mathbf{M}_h V_r(t) = \mathbf{y}_M(t). \quad (2.18)$$

To determine the coefficient vector $\mathbf{a}_{\text{hom}}(t) \in \mathbb{R}^{R_{\text{hom}}}$, we replace $\mathbf{V}_h(t)$ by $\mathbf{V}_r(t)$ in the momentum equation (2.5) and project onto Φ_{hom} ,

$$\begin{aligned} \Phi_{\text{hom}}^T \Omega_h \frac{d}{dt} \mathbf{V}_r(t) &= \frac{d}{dt} \mathbf{a}_{\text{hom}}(t) + \Phi_{\text{hom}}^T \Omega_h \frac{d}{dt} \mathbf{V}_{\text{inhom}}(t) \\ &= \Phi_{\text{hom}}^T F_h^{CD}(\mathbf{V}_r(t), \mathbf{y}_{\text{bc}}(t)) - \Phi_{\text{hom}}^T \mathbf{G}_h \mathbf{p}_h(t). \end{aligned} \quad (2.19)$$

Due to the divergence-gradient duality (2.7) and property (2.17), the pressure term vanishes, so the ODE (2.19) simplifies to

$$\begin{aligned} \frac{d}{dt} \mathbf{a}_{\text{hom}}(t) + \Phi_{\text{hom}}^T \Omega_h \frac{d}{dt} \mathbf{V}_{\text{inhom}}(t) \\ = \Phi_{\text{hom}}^T F_h^{CD}(\Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \mathbf{V}_{\text{inhom}}(t), \mathbf{y}_{\text{bc}}(t)). \end{aligned} \quad (2.20)$$

This ODE does not contain the pressure $\mathbf{p}_h$ anymore, so our ROM is velocity-only, i.e., the pressure is not needed to simulate the velocity. This property is very useful for two reasons: First because we avoid additional computational costs, second because we avoid pressure-related inf-sup-instabilities [9].

The steps taken so far also hold for other choices of lifting functions, e.g., [46, 71]. For the next simplification of ODE (2.20), however, we exploit the specific definition of our lifting function, (2.15), namely the orthogonality property (2.16). Due to property (2.17), this orthogonality property is inherited by the POD basis Φ_{hom} , i.e.,

$$\Phi_{\text{hom}}^T \Omega_h \mathbf{V}_{\text{inhom}}(t) = 0. \quad (2.21)$$

Due to this orthogonality, the left-hand side of (2.20) simplifies to $\frac{d}{dt} \mathbf{a}_{\text{hom}}$, so the ODE for the ROM coefficient $\mathbf{a}_{\text{hom}}(t)$ becomes

$$\frac{d}{dt} \mathbf{a}_{\text{hom}}(t) = \Phi_{\text{hom}}^T F_h^{CD}(\Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \mathbf{V}_{\text{inhom}}(t), \mathbf{y}_{\text{bc}}(t)) \quad (2.22)$$

The initial condition is given by the projection of the FOM initial condition, $\mathbf{a}_{\text{hom}}(0) = \Phi_{\text{hom}}^T \mathbf{V}_h(0)$.

2.3.3 Efficient evaluation of the ODE right-hand side

To simulate our novel ROM, we have to discretize the ODE (2.22) in time and evaluate the right-hand side. As a naive approach, we could first evaluate the FOM-like right-hand side $F_h^{CD}(\Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \mathbf{V}_{\text{inhom}}(t), \mathbf{y}_{\text{bc}}(t))$ and then perform the projection onto Φ_{hom} , both during the online phase of the ROM simulation. However, the computational costs would depend on the FOM dimensions and hence deteriorate the computational complexity of the ROM in the online phase substantially.

Common ways of avoiding such prohibitive costs are hyper-reduction techniques such as the discrete interpolation method (DEIM) [29] and gappy POD [17]. Instead of evaluating the complete FOM-dimensional vector $F_h^{CD}(\Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \mathbf{V}_{\text{inhom}}(t), \mathbf{y}_{\text{bc}}(t))$, these methods only require a small number of entries to construct an approximation of this vector. However, the definition of our lifting function $\mathbf{V}_{\text{inhom}}(t)$ (2.15) contains the inverse $\bar{\mathbf{L}}_h^{-1}$, requiring the solution of a Poisson equation with computational complexity $\mathcal{O}(N_p^2)$. Consequently, sampling a few entries of $\mathbf{V}_{\text{inhom}}(t)$ is already prohibitively expensive.

Instead of applying hyper-reduction to the right-hand side of (2.22) as a whole, we therefore propose to approximate only the boundary condition vector $\mathbf{y}_{\text{bc}}(t)$. In detail, we propose to replace the boundary condition vector $\mathbf{y}_{\text{bc}}(t)$ in the ODE (2.22) and in the definition of the lifting function (2.15) with a linear approximation

$$\tilde{\mathbf{y}}_{\text{bc}}(t) = \Phi_{\text{bc}} \mathbf{a}_{\text{bc}}(t) \approx \mathbf{y}_{\text{bc}}(t), \quad (2.23)$$

with $\Phi_{\text{bc}} \in \mathbb{R}^{N_{\text{bc}} \times R_{\text{bc}}}$, $\mathbf{a}_{\text{bc}}(t) \in \mathbb{R}^{R_{\text{bc}}}$ and $R_{\text{bc}} \ll N_{\text{bc}}$. One option to construct this approximation is to apply hyper-reduction techniques such as DEIM [29]. In that case, the basis Φ_{bc} would be computed via a POD of snapshots of $\mathbf{y}_{\text{bc}}(t)$ during the offline phase. The coefficient vector $\mathbf{a}_{\text{bc}}(t)$ would be computed based on the evaluation of only R_{bc} entries of $\mathbf{y}_{\text{bc}}(t)$ with complexity $\mathcal{O}(R_{\text{bc}}^2)$ at each time step during the online phase. However, such a DEIM coefficient vector is only an approximation to the optimal coefficient. Actually, the best approximation to $\mathbf{y}_{\text{bc}}(t)$ within the space spanned by the basis Φ_{bc} is given by $\Phi_{\text{bc}} \Phi_{\text{bc}}^T \mathbf{y}_{\text{bc}}(t)$. Hence, we can improve the accuracy of the approximation $\tilde{\mathbf{y}}_{\text{bc}}(t)$ by computing the coefficient vector via the projection onto Φ_{bc} , $\mathbf{a}_{\text{bc}}(t) = \Phi_{\text{bc}}^T \mathbf{y}_{\text{bc}}(t)$, instead of performing DEIM. This computation would be prohibitively expensive when performed during the online phase. However, we can maintain the online efficiency by computing $\mathbf{a}_{\text{bc}}(t^j)$ for all required time steps t^j already in the offline phase. This approach requires that $\mathbf{y}_{\text{bc}}(t)$ is known a priori, which we assume in this chapter. If $\mathbf{y}_{\text{bc}}(t)$ would not be known a priori, we suggest to resort to computing $\mathbf{a}_{\text{bc}}(t)$ via DEIM [29].

The replacement of the boundary condition vector $\mathbf{y}_{\text{bc}}(t)$ by the approximation (2.23) introduces an error in the ROM. In particular, the resulting ROM $\mathbf{V}_r(t) = \Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \tilde{\mathbf{V}}_{\text{inhom}}(t)$ does not satisfy the mass equation with respect to the original boundary conditions (2.18), but with respect to the approximated boundary conditions,

$$\mathbf{M}_h \mathbf{V}_r(t) = \tilde{\mathbf{y}}_M(t) := \mathbf{F}_M \tilde{\mathbf{y}}_{\text{bc}}(t). \quad (2.24)$$

The coefficient $\mathbf{a}_{\text{hom}}(t)$ is defined by the ODE

ROM: approximate
lifting function satisfies
approximate boundary
conditions

$$\begin{aligned} & \frac{d}{dt} \mathbf{a}_{\text{hom}}(t) \\ &= \mathbf{\Phi}_{\text{hom}}^T \mathbf{F}_h^{CD} (\mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \tilde{\mathbf{V}}_{\text{inhom}}(t), \mathbf{\Phi}_{\text{bc}} \mathbf{a}_{\text{bc}}(t)), \end{aligned} \quad (2.25)$$

where the approximated lifting function $\tilde{\mathbf{V}}_{\text{inhom}}(t)$ is defined by

$$\tilde{\mathbf{V}}_{\text{inhom}}(t) = \mathbf{\Omega}_h^{-1} \mathbf{G}_h \bar{\mathbf{L}}_h^{-1} \mathbf{F}_M \mathbf{\Phi}_{\text{bc}} \mathbf{a}_{\text{bc}}(t) =: \mathbf{F}_{\text{inhom}} \mathbf{a}_{\text{bc}}(t). \quad (2.26)$$

ROM: approximate lifting function can be computed efficiently

The main advantage of this approximation is the drastic reduction of the number of (regularized) Poisson solves required for computing the lifting function due to the linear dependence on $\mathbf{y}_{\text{bc}}(t)$ in the definition (2.15). Indeed, to precompute the matrix $\mathbf{F}_{\text{inhom}} \in \mathbb{R}^{N_V \times R_{\text{bc}}}$, we need to solve only R_{bc} (regularized) Poisson equations for the R_{bc} columns of the matrix $\mathbf{F}_M \mathbf{\Phi}_{\text{bc}}$.

Note that this approximated lifting function satisfies the same orthogonality property as the exact lifting function, (2.21),

$$\mathbf{\Phi}_{\text{hom}}^T \mathbf{\Omega}_h \tilde{\mathbf{V}}_{\text{inhom}}(t) = \mathbf{0}. \quad (2.27)$$

ROM: approximate lifting function is orthogonal to homogeneous ROM basis

We will use this orthogonality property in Section 2.4 to derive energy consistency.

2.3.4 Exact offline decomposition using matricized third-order tensors

To evaluate the right-hand side of the approximated ODE (2.25) efficiently, we extend the offline precomputation approach in [98]. This approach exploits the low-dimensional structure of the terms $\mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t)$, $\mathbf{F}_{\text{inhom}} \mathbf{a}_{\text{bc}}(t)$ and $\mathbf{\Phi}_{\text{bc}} \mathbf{a}_{\text{bc}}(t)$ as well as the fact that the nonlinearities in $\mathbf{F}_h^{CD}$ are only second order polynomials. In detail, we formulate these second order polynomials as precomputable “matricized” third order tensors. For example, we can write the term quadratic in $\mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t)$ as $\bar{\mathbf{C}}_2 (\mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t) \otimes \mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t))$ with $\bar{\mathbf{C}}_2 \in \mathbb{R}^{N_V \times N_V^2}$ and the Kronecker product $\otimes$. Then, we find

$$\begin{aligned} & \mathbf{\Phi}_{\text{hom}}^T \bar{\mathbf{C}}_2 (\mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t) \otimes \mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t)) \\ &= \mathbf{\Phi}_{\text{hom}}^T \bar{\mathbf{C}}_2 (\mathbf{\Phi}_{\text{hom}} \otimes \mathbf{\Phi}_{\text{hom}}) (\mathbf{a}_{\text{hom}}(t) \otimes \mathbf{a}_{\text{hom}}(t)) \end{aligned} \quad (2.28)$$

where $\mathbf{\Phi}_{\text{hom}}^T \bar{\mathbf{C}}_2 (\mathbf{\Phi}_{\text{hom}} \otimes \mathbf{\Phi}_{\text{hom}}) \in \mathbb{R}^{R_{\text{hom}} \times R_{\text{hom}}^2}$ can be precomputed offline. Similarly, we can precompute operators in $\mathbb{R}^{R_{\text{hom}} \times R_{\text{hom}}^2}$ and $\mathbb{R}^{R_{\text{hom}} \times (R_{\text{hom}} R_{\text{bc}})}$ for the term quadratic in $\mathbf{\Phi}_{\text{bc}} \mathbf{a}_{\text{bc}}(t)$ and the mixed term linear in both, $\mathbf{\Phi}_{\text{hom}} \mathbf{a}_{\text{hom}}(t)$ and $\mathbf{\Phi}_{\text{bc}} \mathbf{a}_{\text{bc}}(t)$, respectively. Hence, the resulting method has an online complexity scaling in $\mathcal{O}(R_{\text{hom}}^3 + R_{\text{hom}}^2 R_{\text{bc}} + R_{\text{hom}} R_{\text{bc}}^2)$.

2.3.5 Pressure recovery

As discussed in Section 2.3.2, our proposed ROM is velocity-only, so no computation of the pressure is needed to simulate the velocity. However, we can compute the pressure $p_r(t)$ of the ROM as a post-processing step by inserting the ROM velocity $V_r(t)$ and the boundary condition approximation $\tilde{y}_{bc}(t)$ into the (regularized) FOM Poisson equation (2.9)

*we can but do not need
to compute the pressure*

$$\tilde{\mathbf{L}}_h p_r(t) = -\frac{d}{dt} \tilde{y}_M(t) + \mathbf{M}_h \Omega_h^{-1} \mathbf{F}_h^{CD}(V_r(t), \tilde{y}_{bc}(t)). \quad (2.29)$$

This pressure is consistent in the sense that the (regularized) Poisson equation (2.29) has the same structure as the (regularized) FOM Poisson equation (2.8). Consequently, the ROM pressure $p_r(t)$ converges to the FOM pressure $p_h(t)$ if the ROM velocity $V_r(t)$ converges to the FOM velocity $V_h(t)$ and the boundary condition approximation $\tilde{y}_{bc}(t)$ converges to the exact boundary condition vector $y_{bc}(t)$.

2.4 KINETIC ENERGY EVOLUTION

A distinctive property of our proposed ROM is its kinetic energy consistency, i.e., the ROM kinetic energy evolution has the same structure as the FOM kinetic energy evolution. We define the FOM kinetic energy as $K_h(t) = \frac{1}{2} \|V_h(t)\|_{\Omega_h}^2$, leading to the kinetic energy evolution

$$\frac{d}{dt} K_h(t) = V_h(t)^T \mathbf{F}_h^{CD}(V_h(t), y_{bc}(t)) + y_M(t)^T p_h(t), \quad (2.30)$$

using the momentum equation (2.5) and the mass equation (2.4).

The key property leading to a kinetic energy evolution with the same structure for our proposed ROM is the orthogonality of $V_{r,\text{hom}}(t)$ and $\tilde{V}_{\text{inhom}}(t)$, given by equation (2.27). Due to this orthogonality the ROM kinetic energy can be decomposed into two terms,

$$K_r(t) = \frac{1}{2} \|V_r(t)\|_{\Omega_h}^2 = \frac{1}{2} \|a_{\text{hom}}(t)\|_2^2 + \frac{1}{2} \|\tilde{V}_{\text{inhom}}(t)\|_{\Omega_h}^2, \quad (2.31)$$

The time-derivative of the first term (the energy of the homogeneous component) is given by

$$\begin{aligned} \frac{d}{dt} \frac{1}{2} \|a_{\text{hom}}(t)\|_2^2 &= a_{\text{hom}}(t)^T \frac{d}{dt} a_{\text{hom}}(t) \\ &= a_{\text{hom}}(t)^T \Phi_{\text{hom}}^T \mathbf{F}_h^{CD}(V_r(t), \tilde{y}_{bc}(t)), \end{aligned} \quad (2.32)$$

using the momentum equation (2.25). Furthermore, the time-derivative of the second term (the energy of the approximated lifting function (2.15)) can be written as

$$\begin{aligned}
\frac{d}{dt} \frac{1}{2} \|\tilde{\mathbf{V}}_{\text{inhom}}(t)\|_{\tilde{\Omega}_h}^2 &= \tilde{\mathbf{V}}_{\text{inhom}}(t)^T \Omega_h \frac{d}{dt} \tilde{\mathbf{V}}_{\text{inhom}}(t) \\
&= \left(\Omega_h^{-1} \mathbf{G}_h \tilde{\mathbf{L}}_h^{-1} \tilde{\mathbf{y}}_M(t) \right)^T \Omega_h \frac{d}{dt} \left(\Omega_h^{-1} \mathbf{G}_h \tilde{\mathbf{L}}_h^{-1} \tilde{\mathbf{y}}_M(t) \right) \\
&= -\tilde{\mathbf{y}}_M(t)^T \tilde{\mathbf{L}}_h^{-T} \mathbf{M}_h \Omega_h^{-1} \mathbf{G}_h \tilde{\mathbf{L}}_h^{-1} \frac{d}{dt} \tilde{\mathbf{y}}_M(t) \\
&= -\tilde{\mathbf{y}}_M(t)^T \tilde{\mathbf{L}}_h^{-1} \frac{d}{dt} \tilde{\mathbf{y}}_M(t), \tag{2.33}
\end{aligned}$$

where we use the divergence-gradient duality (2.7), the symmetry of $\mathbf{L}_h$ and Ω_h , the definition of $\mathbf{L}_h$ (2.8), and the inverse and symmetry properties of $\tilde{\mathbf{L}}_h^{-1}$ (2.10)-(2.11).

Insertion of the Poisson equation for the ROM pressure (2.29) leads to

$$\begin{aligned}
\frac{d}{dt} \frac{1}{2} \|\tilde{\mathbf{V}}_{\text{inhom}}(t)\|_{\tilde{\Omega}_h}^2 &= -\tilde{\mathbf{y}}_M(t)^T \left[-\mathbf{p}_r(t) + \tilde{\mathbf{L}}_h^{-1} \mathbf{M}_h \Omega_h^{-1} \mathbf{F}_h^{CD}(\mathbf{V}_r(t), \tilde{\mathbf{y}}_{\text{bc}}(t)) \right] \\
&= \tilde{\mathbf{y}}_M(t)^T \mathbf{p}_r(t) + \tilde{\mathbf{V}}_{\text{inhom}}(t)^T \mathbf{F}_h^{CD}(\mathbf{V}_r(t), \tilde{\mathbf{y}}_{\text{bc}}(t)). \tag{2.34}
\end{aligned}$$

Here, we have again used the divergence-gradient duality (2.7), the symmetry of $\mathbf{L}_h$ and Ω_h , the symmetry property of $\tilde{\mathbf{L}}_h^{-1}$ (2.11), and identified the definition of the lifting function (2.15),

$$\begin{aligned}
-\tilde{\mathbf{y}}_M(t)^T \tilde{\mathbf{L}}_h^{-1} \mathbf{M}_h \Omega_h^{-1} &= -\tilde{\mathbf{y}}_M(t)^T \tilde{\mathbf{L}}_h^{-T} \mathbf{M}_h \Omega_h^{-1} \\
&= \left(\Omega_h^{-1} \mathbf{G}_h \tilde{\mathbf{L}}_h^{-1} \tilde{\mathbf{y}}_M(t) \right)^T = \tilde{\mathbf{V}}_{\text{inhom}}(t)^T. \tag{2.35}
\end{aligned}$$

Altogether, the kinetic energy evolution of the ROM can be written as

$$\frac{d}{dt} K_r(t) = \mathbf{V}_r(t)^T \mathbf{F}_h^{CD}(\mathbf{V}_r(t), \tilde{\mathbf{y}}_{\text{bc}}(t)) + \tilde{\mathbf{y}}_M(t)^T \mathbf{p}_r(t). \tag{2.36}$$

This ROM energy evolution has exactly the same structure as the FOM energy evolution (2.30), which is our definition of consistency.

This energy consistency will be useful to transfer stability statements from the FOM to the ROM. Namely, if we assume that a bound on the kinetic energy can be derived from the FOM kinetic energy evolution with respect to certain approximated boundary conditions $\tilde{\mathbf{y}}_{\text{bc}}(t)$, then the same derivation could be used to obtain a bound for the ROM kinetic energy. Note, however, that stability (energy) bounds for the incompressible Navier-Stokes equations including time-dependent boundary

FOM and ROM kinetic energy evolutions have same structure

conditions are still an open topic [31, 38], so this is a suggestion for future research.

As indicated above, the orthogonality property (2.27) is key to achieve this energy consistency. In Appendix A.1, we show that a different choice of the lifting function would spoil the energy consistency so that FOM energy bounds could not directly be transferred to the ROM.

To preserve the energy-consistency on the time-discrete level, one could use energy-conserving Runge-Kutta methods [97]. However, in our setting of time-dependent boundary conditions, the kinetic energy is not a conserved quantity and energy-conserving methods do not (and should not) yield a constant energy. Instead, we opt for using high-order explicit methods, which are an accurate and computationally attractive alternative, also given the fact that we are using an exact third-order tensor representation of the convective terms. As shown in Appendix A.2, the difference between explicitly and implicitly integrated ROM velocities is quite small.

2.5 EQUIVALENT VELOCITY-PRESSURE ROM

2.5.1 Velocity-pressure ROM

Our proposed ROM has the appealing property that it is velocity-only. However, we can interpret this ROM as a velocity-pressure ROM, as well. This interpretation provides further insight in the relation between time-dependent boundary conditions and the role of the pressure.

The velocity-pressure ROM that we consider comprises the velocity approximation $V_r(t) = \Phi a(t) \approx V_h(t)$ with the Ω_h -orthonormal basis $\Phi \in \mathbb{R}^{N_V \times R_V}$ and the coefficient vector $a(t) \in \mathbb{R}^{R_V}$, and the pressure approximation $p_r(t) = \Psi b(t) \approx p_h(t)$ with the basis $\Psi \in \mathbb{R}^{N_p \times R_p}$ and the coefficient vector $b(t) \in \mathbb{R}^{R_p}$. In the next section, we will specify the choice of bases such that the resulting velocity-pressure ROM is equivalent to the velocity-only ROM proposed in Section 2.3. In general, however, the bases can also be chosen differently.

We insert these approximations in the mass equation (2.4) projected by the pressure basis Ψ and in the momentum equation (2.5) projected by the velocity basis Φ , leading to:

$$\Psi^T M_h \Phi a(t) = \Psi^T \tilde{y}_M(t) := \Psi^T F_M \tilde{y}_{bc}(t) \quad (2.37)$$

$$\begin{aligned} \Phi^T \Omega_h \Phi \frac{d}{dt} a(t) &= \frac{d}{dt} a(t) = \Phi^T F_h^{CD}(\Phi a(t), \tilde{y}_{bc}(t)) \\ &\quad - \Phi^T G_h \Psi b(t). \end{aligned} \quad (2.38)$$

The initial condition $\mathbf{a}(t^0) = \mathbf{a}^0$ is defined via the constrained optimization problem of minimizing $\|\Phi \mathbf{a}^0 - \mathbf{V}_h(t^0)\|_{\Omega_h}$ such that $\mathbf{a}^0$ satisfies the projected mass equation (2.37) at $t = t^0$.

Analogously to the FOM described in Section 2.2, we solve this system by solving the reduced Poisson equation for $\mathbf{b}(t)$,

$$\begin{aligned} \mathbf{L}_r \mathbf{b}(t) &:= \Psi^T \mathbf{M}_h \Phi \Phi^T \mathbf{G}_h \Psi \mathbf{b}(t) \\ &= -\Psi^T \frac{d}{dt} \tilde{\mathbf{y}}_M(t) + \Psi^T \mathbf{M}_h \Phi \Phi^T \mathbf{F}_h^{CD}(\Phi \mathbf{a}(t), \tilde{\mathbf{y}}_{bc}(t)). \end{aligned} \quad (2.39)$$

Then, due to the pressure term in the ODE (2.38), $\mathbf{a}(t)$ computed from this ODE is guaranteed to satisfy the projected mass equation (2.37). In contrast to the FOM, the system matrix $\mathbf{L}_r$ of this reduced Poisson equation is not high-dimensional and sparse, but low-dimensional and dense. Hence, the system is solved (by employing a precomputed LU-factorization) with a computational complexity of $\mathcal{O}(R_p^2)$ at each time step in the online phase. This approach requires the invertibility of the system matrix $\mathbf{L}_r = \Psi^T \mathbf{M}_h \Phi \Phi^T \mathbf{G}_h \Psi$. We will guarantee this invertibility by choosing the bases Φ and Ψ suitably, as explained in the next section.

Analogously to the discussion in Section 2.3.3 for the velocity-only ROM, computing the right-hand side of the ODE (2.38) for the exact boundary condition vector $\mathbf{y}_{bc}(t)$ would in general be prohibitively expensive. Hence, we have replaced the exact boundary condition vector $\mathbf{y}_{bc}(t)$ in (2.37)-(2.39) by the lower-dimensional approximation $\tilde{\mathbf{y}}_{bc}(t) := \Phi_{bc} \mathbf{a}_{bc}(t)$ as described in Section 2.3.3.

2.5.2 Equivalence to the proposed velocity-only ROM

The following proposition states that the velocity-pressure ROM (2.37)-(2.38) is equivalent to the velocity-only ROM (2.25)-(2.26) if we choose the bases Φ and Ψ appropriately.

Proposition 2.1 (Equivalence of velocity-only and velocity-pressure ROM).

If the Ω_h -orthonormal velocity basis $\Phi \in \mathbb{R}^{N_V \times R_V}$ and the pressure basis $\Psi \in \mathbb{R}^{N_p \times R_p}$ are chosen such that

- a) the image space of Φ is equivalent to the union of the image spaces of Φ_{hom} in equation (2.17) and $\mathbf{F}_{\text{inhom}}$ as defined in equation (2.26).
- b) for all $t \in [0, T]$, there exists a solution to the unprojected mass equation

$$\mathbf{M}_h \Phi \mathbf{a}(t) = \tilde{\mathbf{y}}_M(t), \quad (2.40)$$

- c) $R_p = \text{rank}(\mathbf{M}_h \Phi)$,

The velocity-only ROM is equivalent to a specific velocity-pressure ROM.

d) and $\Psi^T \mathbf{M}_h \Phi$ has rank R_p ,

then the velocity of the velocity-pressure ROM, $\mathbf{V}_{r,\text{velocity-pressure}}(t) = \Phi \mathbf{a}(t)$, is for all $t \in [0, T]$ equivalent to the velocity of the velocity-only ROM, $\mathbf{V}_{r,\text{velocity-only}}(t) = \Phi_{\text{hom}} \mathbf{a}_{\text{hom}}(t) + \tilde{\mathbf{V}}_{\text{inhom}}(t)$ described in Section 2.3.

Proof. First of all, Assumption (a) implies that there exists an orthonormal matrix $\mathbf{Q} \in \mathbb{R}^{R_V \times R_V}$ such that $\Phi = [\Phi_{\text{hom}} \quad \Phi_{\text{inhom}}] \mathbf{Q}$. Here, $\Phi_{\text{inhom}} \in \mathbb{R}^{N_V \times R_{\text{inhom}}}$ with $R_{\text{inhom}} = \text{rank}(\mathbf{F}_{\text{inhom}})$ is an Ω_h -orthonormal basis with the same image space as $\mathbf{F}_{\text{inhom}}$ which can be computed from $\mathbf{F}_{\text{inhom}}$, e.g., via a Gram-Schmidt-orthonormalization. We can decom-

pose the matrix $\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_1 \\ \mathbf{Q}_2 \end{bmatrix}$ with $\mathbf{Q}_1 \in \mathbb{R}^{R_{\text{hom}} \times R_V}$ and $\mathbf{Q}_2 \in \mathbb{R}^{R_{\text{inhom}} \times R_V}$.

Then, we write the velocity-pressure ROM velocity as

$$\begin{aligned} \mathbf{V}_{r,\text{velocity-pressure}}(t) &= \Phi \mathbf{a}(t) \\ &= \Phi_{\text{hom}} \mathbf{Q}_1 \mathbf{a}(t) + \Phi_{\text{inhom}} \mathbf{Q}_2 \mathbf{a}(t) \\ &=: \Phi_{\text{hom}} \mathbf{a}_1(t) + \Phi_{\text{inhom}} \mathbf{a}_2(t). \end{aligned} \quad (2.41)$$

We proceed to show that the velocity components of the two ROMs are equivalent, namely

$$\mathbf{a}_1(t) = \mathbf{a}_{\text{hom}}(t) \quad \text{for all } t \in [0, T], \quad (2.42)$$

and

$$\Phi_{\text{inhom}} \mathbf{a}_2(t) = \tilde{\mathbf{V}}_{\text{inhom}}(t) \quad \text{for all } t \in [0, T]. \quad (2.43)$$

For this purpose, we note that the decomposition (2.41) separates the modes in Φ_{hom} , which lie in the kernel of $\mathbf{M}_h$, from those in Φ_{inhom} , which lie outside the kernel of $\mathbf{M}_h$. Consequently, equations (2.37) and (2.40) simplify to

$$\Psi^T \mathbf{M}_h \Phi_{\text{inhom}} \mathbf{a}_2(t) = \Psi^T \tilde{\mathbf{y}}_M(t), \quad (2.44)$$

and

$$\mathbf{M}_h \Phi_{\text{inhom}} \mathbf{a}_2(t) = \tilde{\mathbf{y}}_M(t), \quad (2.45)$$

respectively. Let, for a fixed t , $\hat{\mathbf{a}}_2$ and $\bar{\mathbf{a}}_2$ be the solutions to (2.44) and (2.45), respectively (their existence follows from Assumption (b)). Then, premultiplying (2.45) by Ψ^T , we find

$$\Psi^T \mathbf{M}_h \Phi_{\text{inhom}} \bar{\mathbf{a}}_2 = \Psi^T \mathbf{y}_M(t) = \Psi^T \mathbf{M}_h \Phi_{\text{inhom}} \hat{\mathbf{a}}_2. \quad (2.46)$$

By construction of Φ_{inhom} , we have $\text{rank}(\mathbf{M}_h \Phi) = \text{rank}(\mathbf{M}_h \Phi_{\text{inhom}}) = R_{\text{inhom}}$. Because of Assumption (c), $R_p = R_{\text{inhom}}$, and because of Assumption (d), $\text{rank}(\Psi^T \mathbf{M}_h \Phi_{\text{inhom}}) = \text{rank}(\Psi^T \mathbf{M}_h \Phi) = R_{\text{inhom}}$. Hence, $\Psi^T \mathbf{M}_h \Phi_{\text{inhom}} \in \mathbb{R}^{R_{\text{inhom}} \times R_{\text{inhom}}}$ is invertible, so (2.46) implies $\bar{a}_2 = \hat{a}_2$. Hence, the solution $\hat{a}_2$ to the projected mass equation (2.44) also satisfies the unprojected mass equation (2.45). This solution $\hat{a}_2$ to the unprojected mass equation is unique. To show this uniqueness, let $\check{a}_2$ be a second solution to (2.45). Then, we find

$$\mathbf{M}_h \Phi_{\text{inhom}}(\hat{a}_2 - \check{a}_2) = \mathbf{y}_M(t) - \mathbf{y}_M(t) = \mathbf{0}. \quad (2.47)$$

Since $\mathbf{M}_h \Phi_{\text{inhom}}$ has full column rank, we infer $\hat{a}_2 - \check{a}_2 = \mathbf{0}$, so the two solutions are equivalent. This uniqueness implies the equivalence of the inhomogeneous velocity components (2.43) because $\mathbf{F}_{\text{inhom}}$ has the same image space as Φ_{inhom} and both terms satisfy the unprojected mass equation for all $t \in [0, T]$.

What remains to be shown is the equivalence of the homogeneous velocity components (2.42). For this purpose, we premultiply the momentum equation (2.38) by $\mathbf{Q}_1$ to find

$$\begin{aligned} \frac{d}{dt} \mathbf{Q}_1 \mathbf{a}(t) &= \frac{d}{dt} \mathbf{a}_1(t) \\ &= \Phi_{\text{hom}}^T \mathbf{F}_h^{CD} (\Phi_{\text{hom}} \mathbf{a}_1(t) + \tilde{\mathbf{V}}_{\text{inhom}}(t), \tilde{\mathbf{y}}_{\text{bc}}(t)), \end{aligned} \quad (2.48)$$

where $\mathbf{Q}_1 \Phi^T = \Phi_{\text{hom}}^T$ and the pressure term $\Phi_{\text{hom}}^T \mathbf{G}_h \Psi \mathbf{b}(t)$ disappears due to equation (2.17). This ODE is equivalent to the velocity-only ODE (2.25). As both ODEs have the same initial condition, they also have the same solution, so we find (2.42).

Altogether, we hence find

$$\mathbf{V}_{r, \text{velocity-pressure}}(t) = \mathbf{V}_{r, \text{velocity-only}}(t) \quad \text{for all } t \in [0, T]. \quad (2.49)$$

□

The requirements of the proposition are met, for example, by the choice $\Phi = [\Phi_{\text{hom}} \ \Phi_{\text{inhom}}]$ and $\Psi = \mathbf{M}_h \Phi_{\text{inhom}}$. For this choice, we have $\mathbf{Q} = \mathbf{I}$. Note that this choice also guarantees the invertibility of $\mathbf{L}_r$.

The proof of the proposition provides an explanation why eliminating the pressure in the velocity-only ROM is possible. As can be seen from ODEs (2.38) and (2.48), the pressure term does not affect the coefficients of the homogeneous modes ($\mathbf{a}_1(t)$) but only those of the inhomogeneous modes ($\mathbf{a}_2(t)$). Since these inhomogeneous coefficients are uniquely determined by the mass equation (2.37), they do not require simulating ODE (2.38), so the pressure is superfluous. This observation is in line with the common understanding that the pressure is a (vector of) Lagrange multiplier(s) enforcing the velocity to satisfy

the mass equation. This understanding also implies that the pressure does not affect the velocity beyond enforcing the mass equation. Indeed, the error between the approximated pressure $p_r(t) = \Psi b(t)$ and the FOM pressure $p_h(t)$ does not introduce an additional error in the ROM velocity.

In practice, the velocity-pressure formulation is slightly more expensive than the velocity-only formulation because the ODE system is integrated in time for $R_V = R_{\text{hom}} + R_{\text{inhom}}$ instead of only R_{hom} coefficients and additionally the pressure coefficients have to be computed by solving the reduced Poisson equation (2.39) at every time step. Hence, the ROM described in Section 2.3 remains the method of choice.

Analogously to [4], we can decompose the velocity-pressure ROM velocity into a homogeneous and an inhomogeneous component using a QR decomposition for arbitrary choices of Φ and Ψ provided that L_r is invertible. Provided the discretization satisfies the gradient-divergence duality (2.7), this decomposition can be formulated as a velocity-only ROM. Due to the stability problems of the QR decomposition described in [4], this velocity-only ROM would in general not be useful in practice.

2.5.3 Comparison with standard POD velocity-pressure ROMs

A common choice for the velocity basis Φ in the velocity-pressure ROM defined by (2.37) and (2.38) is the ordinary POD basis. The resulting ROM does not in general meet the requirements of Proposition 2.1 because the POD basis does not necessarily satisfy Assumption (b). Consequently, this standard POD velocity-pressure ROM is in general not equivalent to the proposed velocity-only ROM. In fact, such a standard POD velocity-pressure ROM can suffer from inf-sup instabilities [105] whereas the proposed velocity-only ROM cannot.

On the other hand, the important insight that Proposition 2.1 does give, is that the proposed velocity-only ROM is equivalent to a velocity-pressure ROM, which differs from the standard POD velocity-pressure ROM only in the choice of velocity basis Φ (and possibly pressure basis Ψ). To compare these differing velocity bases, let us define the best approximation error over the snapshots V_h^j ,

$$\varepsilon(\phi) = \sum_{j=1}^K \left\| V_h^j - \phi \phi^T \Omega_h V_h^j \right\|_{\Omega_h}^2. \quad (2.50)$$

Then, the standard POD bases with $R_V = R_{\text{hom}}$ and $R_V = R_{\text{hom}} + R_{\text{bc}}$ modes, $\Phi_{[R_{\text{hom}}]}$ and $\Phi_{[R_{\text{hom}} + R_{\text{inhom}}]}$ satisfy the relation

$$\varepsilon \left(\Phi_{[R_{\text{hom}} + R_{\text{inhom}}]} \right) \leq \varepsilon \left([\Phi_{\text{hom}} \quad \Phi_{\text{inhom}}] \right) \leq \varepsilon \left(\Phi_{[R_{\text{hom}}]} \right), \quad (2.51)$$

Best approximation errors of velocity-only and velocity-pressure ROMs are comparable.

assuming exact boundary conditions $\tilde{y}_{bc}(t) = y_{bc}(t)$. The proof can be found in Appendix A.3.

If the ROM errors are dominated by these best approximation errors, we can consequently expect the accuracy of the proposed velocity-only ROM to lie in between the standard POD velocity-pressure ROMs with $R_V = R_{\text{hom}} + R_{\text{inhom}}$ and with $R_V = R_{\text{hom}}$ modes, provided that the standard POD velocity-pressure ROMs are stable.

As explained in the previous section, velocity-pressure ROMs with $R_V = R_{\text{hom}} + R_{\text{inhom}}$ modes are slightly more expensive than the velocity-only ROM with R_{hom} and R_{bc} modes. Also, velocity-pressure ROMs with $R_V = R_{\text{hom}}$ modes are slightly more expensive than the velocity-only ROM due to the need to solve Poisson equations for the pressure.

The accuracy per computation time of the proposed velocity-only ROM is consequently similar to standard POD velocity-pressure ROMs. While the latter are known to be prone to instabilities, the velocity-only ROM is energy-consistent and does not suffer from inf-sup instability. Furthermore, while standard POD velocity-pressure ROMs only satisfy a projected mass equation, the velocity-only ROM satisfies the unprojected mass equation with respect to explicitly approximated boundary conditions.

2.6 NUMERICAL EXPERIMENTS

In our numerical experiments, we investigate our proposed velocity-only ROM regarding convergence behaviour, mass conservation and energy consistency. Furthermore, we verify the equivalence to the velocity-pressure ROM described in Section 2.5.2 and analyse the runtime.

2.6.1 Setup

2.6.1.1 Testcases

We consider two testcases based on [97, Section 6.4.2] modelling air flow around a wind turbine. The simulation domain is $[0, 10] \times [-2, 2]$ with an actuator disk of length 1 at $x = 2$ as depicted in Fig. 2.2. On the top ($y = y_{\text{max}} = 2$), on the bottom ($y = y_{\text{min}} = -2$) and on the right ($x = 10$), we prescribe outflow boundary conditions

$$(-pI + \nu \nabla u) \cdot n = -p_\infty I \cdot n \quad (2.52)$$

with $p_\infty = 0$ and $\nu = 10^{-2} \frac{m^2}{s}$. In contrast to [97, Section 6.4.2], we consider inflow on the left boundary ($x = 0$) which is not only time- but

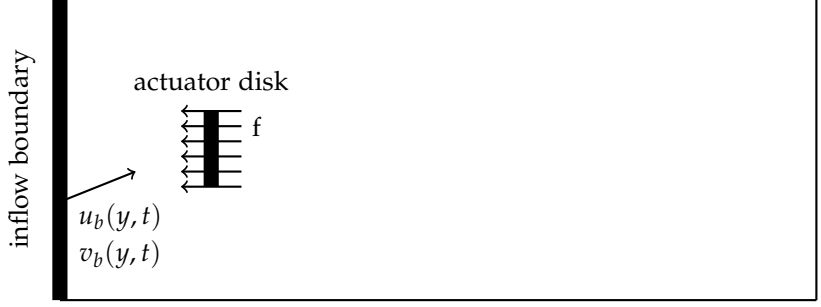


Figure 2.2: Sketch of testcase setup with actuator disk and time- and space-dependent boundary conditions.

also space-dependent. For the first testcase *varying angle*, we consider inflow with constant magnitude 1 but time- and space-dependent angle,

$$u_b(y, t) = \cos(\alpha(y, t)), \quad v_b(y, t) = \sin(\alpha(y, t)),$$

$$\text{with } \alpha(y, t) = \frac{\pi}{6} \sin\left(y - \frac{t}{2}\right). \quad (2.53)$$

As we will see in Section 2.6.1.3, these boundary conditions can be approximated well by a lower dimensional approximation $\tilde{y}_{bc}(t)$ as described in Section 2.3.3. In contrast, the second testcase exemplifies the extreme and in practice unlikely case that the boundary conditions cannot be reduced effectively. For this second testcase *moving mode*, we move the parabolic inflow field $u_{\text{parabolic}}(y) = \frac{1}{10}(y - y_{\min})(y_{\max} - y)$ uniformly such that it enters the simulation domain from the bottom at t_{start} and leaves it at the top at t_{end} ,

$$u_b(y, t) = u_{\text{parabolic}}\left(y + \frac{t - t_{\text{end}}}{t_{\text{end}} - t_{\text{start}}}(y_{\max} - y_{\min})\right)$$

$$\text{for } t_{\text{start}} \leq t \leq t_{\text{end}}. \quad (2.54)$$

This inflow does not have a tangential component, so $v_b = 0$.

For both testcases, the Poisson matrix $\mathbf{L}_h$ is invertible, so $\bar{\mathbf{L}}_h = \mathbf{L}_h$ in the FOM Poisson equation (2.8) and in the definition of the lifting function (2.26).

The actuator disk represents a momentum sink and is modeled as a constant force $f = 0.25$ acting in negative x -direction and we consider the time intervals $[0, 4\pi]$ and $[0, 20]$, respectively.

2.6.1.2 Numerical methods

We discretize the simulation domain with a uniform 200×80 staggered grid. For time integration, we use the explicit fourth-order Runge-Kutta

method RK4 [21] and discretize for both testcases the respective time interval equidistantly with time step size $\Delta t = \frac{t_{\text{end}}}{800}$.

The initial condition is given by $V_h^0 = V_{\text{inhom}}^0$ which is computed according to (2.15) based on the respective boundary conditions. For the *moving mode* testcase, this definition results in $V_h^0 = \mathbf{0}$ while the initial condition for the *varying angle* testcase is a nonzero function.

The POD bases Φ_{hom} and Φ_{bc} are constructed from the snapshot matrices

$$\begin{aligned} \mathbf{X}_{\text{hom}} &= \begin{bmatrix} V_{h,\text{hom}}^0 & \dots & V_{h,\text{hom}}^j & \dots & V_{h,\text{hom}}^{800} \end{bmatrix} \quad \text{and} \\ \mathbf{X}_{\text{bc}} &= \begin{bmatrix} y_{\text{bc}}(t^0) & \dots & y_{\text{bc}}(t^j) & \dots & y_{\text{bc}}(t^{800}) \end{bmatrix}, \end{aligned} \quad (2.55)$$

where $V_{h,\text{hom}}^j$ is the time discrete approximation of $V_{h,\text{hom}}(t^j)$ with $t^j = j \cdot \Delta t$.

We compute the coefficient vector $\mathbf{a}_{\text{bc}}(t) = \Phi_{\text{bc}} \Phi_{\text{bc}}^T y_{\text{bc}}(t)$ during the offline precomputation phase for all required time steps, as described in Section 2.3.3.

The numbers of ROM modes R_{hom} and R_{bc} can be chosen independently of each other. In the numerical experiments here, however, we choose $R_{\text{hom}} = R_{\text{bc}} = R$ for convenience only, and consider velocity-only ROMs for different values of R .

2.6.1.3 Singular value decays

First, we investigate the singular value decays of the snapshot matrices which we construct our ROM bases from. As can be seen in Fig. 2.3a, only 23 singular values of the boundary condition vector snapshot matrix $\mathbf{X}_{\text{bc}}$ have a significant magnitude for the *varying angle* testcase. In fact, only 5 modes are required to capture at least 99% of the snapshot energy [49].

In contrast, Fig. 2.3b shows that 80 singular values of $\mathbf{X}_{\text{bc}}$ for the *moving mode* testcase, which we designed to have irreducible boundary conditions, have a relative magnitude larger than 10^{-4} . This number equals the number of grid points at the inflow boundary and the number of nonzero entries in $\mathbf{y}_M$, so these entries do not exhibit any lower-dimensional coherent structure. Here, 29 modes are required to capture at least 99% of the snapshot energy.

Similarly, the decays of singular values of the velocity snapshot matrices $\mathbf{X}_{\text{hom}}$ shown in Fig. 2.4 differ considerably for large mode numbers. While only 164 singular values are larger than machine precision (when divided by the largest singular value), for the *varying angle* testcase, 684 singular values are larger than this relative threshold for the *moving mode* testcase. For small mode numbers, however, both decays are

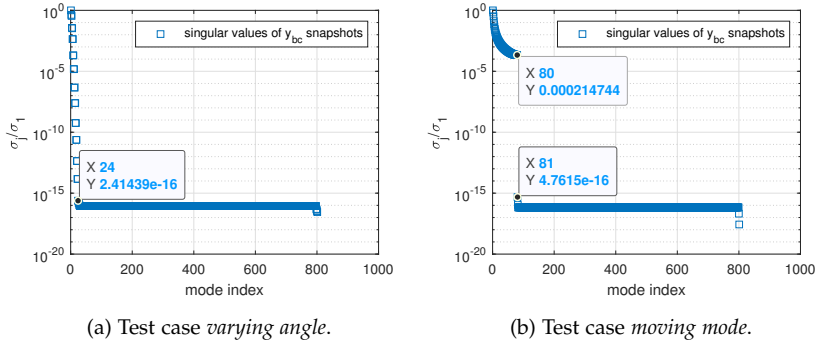


Figure 2.3: Singular values of the snapshot matrices X_{bc} for the two testcases divided by largest singular value.

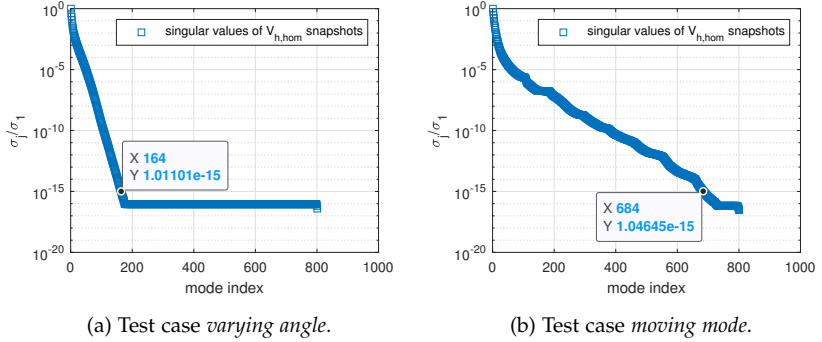


Figure 2.4: Singular values of the snapshot matrices X_{hom} for the two testcases.

similarly fast, and only 13 and 14 modes capture at least 99% of the snapshot energy, respectively.

2.6.2 Results

2.6.2.1 Boundary condition approximation accuracy

Before investigating the proposed ROM itself, we investigate the accuracy of the boundary condition approximation for the two test cases. As can be seen in Fig. 2.5, the best approximations are accurate up to machine precision for $R_{bc} \geq 23$ and $R_{bc} \geq 80$, respectively. This observation is perfectly in line with Fig. 2.3 that shows 23 and 80 significant singular values. For the approximations via DEIM, we observe accuracy up to machine precision for $R_{bc} = 23$ and $R_{bc} = 80$, but large errors if the numbers of modes are increased further. These large errors are likely due to the fact that these numbers of modes exceed the rank of

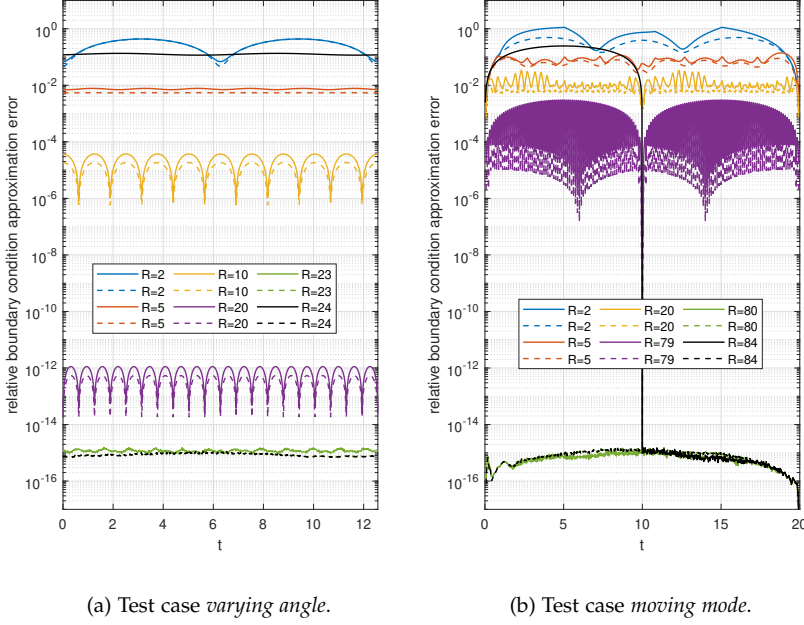


Figure 2.5: Relative boundary condition vector approximation error $\frac{\|\tilde{y}_{bc}(t^j) - y_{bc}(t^j)\|_{\Omega_h}}{\text{average}_j[\|y_{bc}(t^j)\|_{\Omega_h}]}$ for the two test cases for boundary condition DEIM approximations (solid lines) and best approximations (dashed lines) with $R_{bc} = R$ modes.

the corresponding snapshot matrix, causing high sensitivity to machine precision errors. When using the DEIM approach, it is consequently important to know the rank of the snapshot matrix X_{bc} . In the following, we set $R_{bc} = \min(R, 23)$ for the test case *varying angle* and $R_{bc} = R$ for the test case *moving mode*.

2.6.2.2 Convergence

First, we investigate whether the ROM converges to the FOM for increasing number of modes. As can be seen in Fig. 2.6 the relative velocity error clearly decreases monotonously for increasing number of ROM modes. Consequently, our velocity-only ROM for time-dependent boundary conditions indeed provides a convergent approximation to the FOM for both boundary condition approximation approaches.

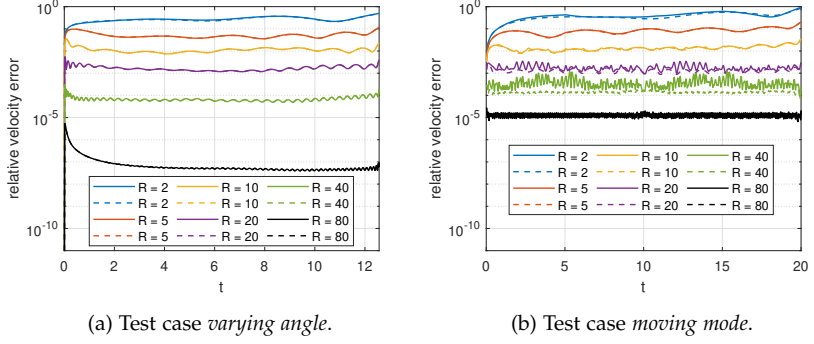


Figure 2.6: Relative velocity error $\frac{\|V_h^j - V_r^j\|_{\Omega_h}}{\text{average}_j[\|V_h^j\|_{\Omega_h}]}$ for the two testcases for boundary condition DEIM approximations (solid lines) and best approximations (dashed lines).

2.6.2.3 Mass conservation

Second, we investigate to which extent our proposed ROM violates the original mass equation (2.18). As explained in Section 2.3.3, the proposed ROM satisfies the mass equation exactly, but with respect to the approximated boundary conditions. The difference between the two is shown in Fig. 2.7, effectively indicating how well the mass equation right-hand side $y_M(t)$ is approximated by $\tilde{y}_M(t)$.

Indeed, as can be seen in Fig. 2.7 the mass conservation violation decreases for increasing number of boundary condition approximation modes. For the testcase *varying angle*, the mass conservation violation is in the order of machine precision for $R \geq 20$. In contrast, for the *moving mode* testcase, $R = 80$ modes are required to achieve a similar mass conservation accuracy. Even for $R = 40$ modes, the mass conservation violation is larger than 10^{-4} . This observation can directly be explained by the singular value decay depicted in Fig. 2.3b, which shows that the 80 largest singular values all have a relative magnitude larger than 10^{-4} . Note however, that in this case the boundary conditions were chosen on purpose to be highly ‘irreducible’ with POD and as such this can be seen as a limiting case, which is very unlikely in practice. Furthermore, the violation of the exact mass equation seems not to impair the accuracy of the actual solution V_r , as can be observed by comparing Fig. 2.6a to 2.6b.

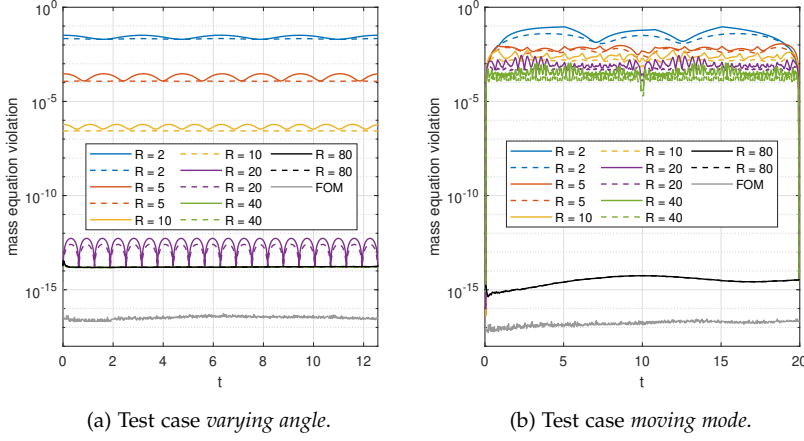


Figure 2.7: Mass conservation violation $\|\mathbf{M}_h \mathbf{V}_r^j - \mathbf{y}_M(t^j)\|_2$ for the two testcases for boundary condition DEIM approximations (solid lines) and best approximations (dashed lines).

2.6.2.4 Energy consistency

Third, we investigate whether the structural correspondence of the FOM and ROM kinetic energy evolutions derived in Section 2.4 is accompanied by small energy errors in practice.

Indeed, Fig. 2.8 shows that the relative kinetic energy errors are approximately at the same order of magnitude as the corresponding relative velocity errors depicted in Fig. 2.6.

2.6.2.5 Verification of ROM equivalence

Next, we verify the ROM equivalence demonstrated in Section 2.5.2. For this purpose, we simulate the same testcases as in the previous experiment but now with the velocity-pressure ROM (2.37)-(2.38) with the bases $\Phi = [\Phi_{\text{hom}} \ \Phi_{\text{inhom}}]$ and $\Psi = \mathbf{M}_h \Phi_{\text{inhom}}$ as described in Section 2.5.2. Here, we compute Φ_{inhom} via a QR decomposition of $\mathbf{F}_{\text{inhom}}$. Hence, the numbers of modes of the velocity-pressure ROM are given by $R_{\text{bc}} = R_{\text{inhom}} = R_p = R$ and $R_V = R_{\text{hom}} + R_{\text{inhom}} = 2R$. As can be seen in Fig. 2.9, the error between the two ROMs ranges for all values of R between 10^{-18} and 10^{-10} . Hence, the ROM equivalence is confirmed up to machine precision.

2.6.2.6 Runtime analysis

Fig. 2.10 shows the computation time of the ROM in comparison with the FOM. For all considered numbers of modes, we observe a significant

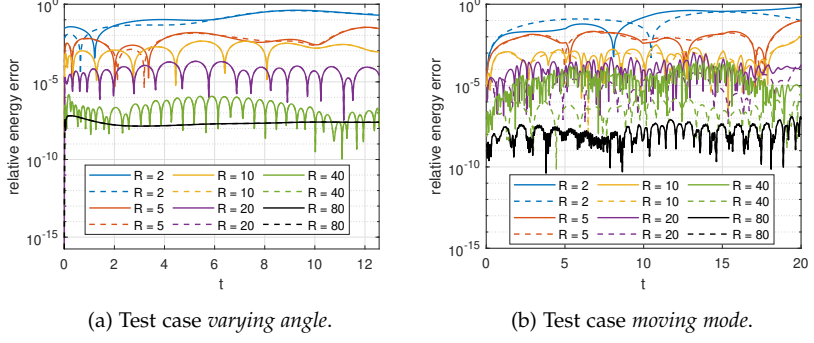


Figure 2.8: Relative kinetic energy error $\frac{|K_h^j - K_r^j|}{\text{average}_j[K_h^j]}$ for both testcases for boundary condition DEIM approximations (solid lines) and best approximations (dashed lines).

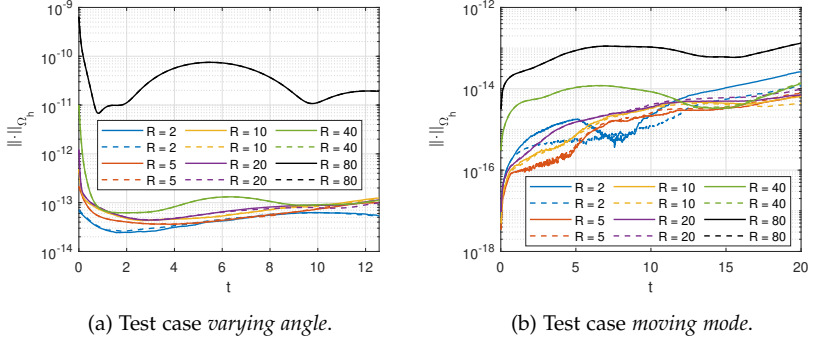


Figure 2.9: ROM equivalence error $\left\| \mathbf{V}_{r,\text{velocity-only}}^j - \mathbf{V}_{r,\text{velocity-pressure}}^j \right\|_{\Omega_h}$ for both testcases for boundary condition DEIM approximations (solid lines) and best approximations (dashed lines).

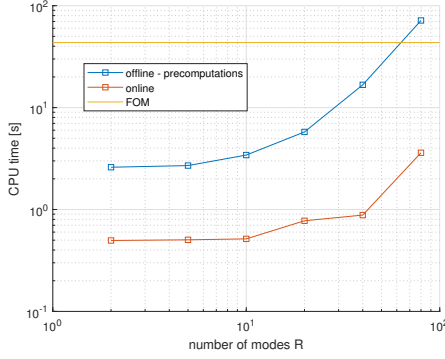


Figure 2.10: Computation time for testcase *moving mode* with $R_{\text{hom}} = R_{\text{inhom}} = R_{\text{bc}} = R$.

speed-up of the ROM online phase compared to the FOM simulations. As in [98], this speed-up amounts to approximately two orders of magnitude for moderate numbers of modes. The offline precomputations, on the other hand, have significant costs and even exceed the FOM simulation costs for $R = 80$.

However, when used in practice for a parametric study, the ROM would already pay off for a small number of ROM simulations.

2.7 CONCLUSION

In this chapter we have generalized the energy-conserving ROM for the incompressible Navier-Stokes equations proposed by Sanderse [98] to arbitrary time-dependent boundary conditions. Our ROM is velocity-only, which avoids computing the pressure and pressure-related inf-sup instability, and it preserves the structure of the kinetic energy evolution. The ROM is based on a Helmholtz-Hodge-inspired decomposition that separates the velocity into two orthogonal components: i) a velocity that satisfies the mass equation with homogeneous boundary conditions and ii) a lifting function that satisfies the mass equation with the prescribed time-dependent boundary conditions. The homogeneous component is defined by the Galerkin-projected momentum equation. Due to the divergence-free projection basis, the pressure term disappears from the momentum equation resulting in the proposed velocity-only ROM.

To allow efficient simulation of the ROM, we approximate the prescribed boundary conditions by a lower-dimensional approximation via POD. The resulting ROM satisfies the mass equation with respect to these approximated boundary conditions. Our numerical experiments show that the error introduced by this boundary condition approxi-

mation does not dominate the velocity error. In these experiments, we use as many modes to approximate the boundary conditions as we use to approximate the homogeneous velocity field. In future research, different choices of these mode numbers should be investigated as well.

We have theoretically shown that the orthogonality of the components leads to a ROM kinetic energy evolution that has the same structure as the FOM kinetic energy evolution. Our numerical experiments confirm that the proposed ROM is able to accurately mimic the kinetic energy evolution. In future work, the structural equivalence of the kinetic energy evolution can be used to derive bounds on the kinetic energy and hence to derive nonlinear stability results.

Furthermore, we have proven the important result that the proposed velocity-only ROM is equivalent to a velocity-pressure ROM, when employing a specific choice of basis functions, and confirmed this equivalence in our numerical experiments. This equivalence can function as a key concept to understand the role of the pressure in ROMs for the incompressible Navier-Stokes equations. It gives important insights into the behaviour of the velocity-pressure ROM regarding the choice for the number of velocity, pressure and boundary condition modes. While this ROM equivalence is based on the assumption of the discrete gradient-divergence duality, future research should investigate whether discretizations that do not exhibit this property also allow for a similar equivalence statement.

In numerical experiments, we have shown the convergence of the ROM upon increasing the number of ROM modes and the accuracy of the boundary condition approximation. Instead of performing a hyper-reduction technique for the convective term, we have used an exact offline decomposition. Scaling cubically in the number of modes, this offline decomposition is limited to small numbers of modes. For moderate numbers of modes, the speed-up of the ROM online phase compared to the FOM is approximately two orders of magnitude. If larger numbers of modes are required that prohibit this cubic scaling, we suggest to incorporate energy-conserving hyper-reduction methods [61].

In future work, it should be investigated how the proposed method performs in practically relevant testcases involving higher Reynolds numbers and generalization over parameter spaces, varying boundary and initial conditions as well as longer time intervals. In case of higher Reynolds numbers, it might be useful to combine our approach with methods that improve the representation of dissipative small scales [7, 8]. For such combinations, we suggest to keep the lifting function unchanged and only adapt the construction of the homogeneous velocity approximation.

EXACT OPERATOR INFERENCE WITH MINIMAL DATA

This chapter presents an advance in the field of non-intrusive model reduction. A novel method is introduced to generate snapshot data for operator inference that guarantees the exact reconstruction of intrusive projection-based ROMs.

To ensure exact reconstruction, the operator inference least squares matrix must have full rank, without regularization. Existing works have achieved this full rank using heuristic strategies to generate snapshot data and a-posteriori checks on full rank, but without a guarantee of success. The novel snapshot data generation method provides this guarantee thanks to two key ingredients: first we identify ROM states that induce full rank, then we generate snapshots corresponding to exactly these states by simulating multiple trajectories for only a single time step. This way, the number of required snapshots is minimal and orders of magnitude lower than typically reported with existing methods. The method avoids non-Markovian terms and does not require re-projection. Since the number of snapshots is minimal, the least squares problem simplifies to a linear system that is numerically more stable. In addition, because the inferred operators are exact, properties of the intrusive ROM operators such as symmetry or skew-symmetry are preserved.

Numerical results for differential equations involving 2nd, 3rd and 8th order polynomials demonstrate that the novel snapshot data generation method leads to exact reconstruction of the intrusive reduced order models.

3.1 INTRODUCTION

Computational efficiency is of paramount importance in complex engineering tasks such as optimization, data assimilation, uncertainty quantification and the development of digital twins. All these tasks require multiple simulations with similar settings. While first principle-derived models have primarily been developed to run single simulations, data-driven models are being developed to exploit the similarity among such multiple simulations.

These data-driven surrogate models vary in how much they preserve structure of the corresponding first principle-derived model [63]. Statistical data-fit models such as Gaussian process models [117] generally do not preserve such structure and solely approximate the mapping

This chapter is based on the following article:
 Henrik Rosenberger,
 Benjamin Sanderse, and
 Giovanni Stabile. "Exact
 operator inference with
 minimal data." In: SIAM
 Journal on Scientific
 Computing 48.5 (2026),
 A2391–C1160. DOI:
 10.1137/25M177248.

between simulation inputs and outputs. Reduced-order models (ROMs), on the other hand, are closely based on first principle-derived, full-order models (FOMs) and perform simulations on a lower-dimensional subspace, for example, by projecting the first principle-derived model onto this subspace [5, 56, 95]. Advantages of such projection-based ROMs are the availability of rigorous theory that provides a posteriori error estimates [52, 57, 85, 109, 111], and the smaller demand for training data compared with surrogate models that preserve less structure. A disadvantage of projection-based ROMs is that they are intrusive in the sense that they require access to the numerical operators of the FOM to perform the projection onto the lower-dimensional subspace. This access is often not possible in software used in industry, and hence impedes the use of intrusive reduced-order models in practice.

To address this practical problem, methods have been developed to approximate intrusive reduced models in a non-intrusive way. These non-intrusive methods do not require access to the numerical operators of the FOM, but only to the generated output data. For linear models, dynamic mode decomposition finds the linear operator that fits the data best [65]. For nonlinear models, operator inference generalizes this approach to polynomial operators [78]. For non-polynomial models, a variable transformation can be used to obtain a polynomial model [86], and so operator inference can still be applied.

However, two major difficulties currently impede operator inference in accurately reconstructing the intrusive ROM operators. First, the FOM snapshot data contains non-Markovian information that is inconsistent with the Markovian intrusive ROM. This non-Markovian information perturbs the operator inference minimization problem such that the intrusive ROM generally does not minimize this perturbed problem. To eliminate this perturbation, Peherstorfer has proposed to generate additional FOM snapshot data whereby the FOM state is projected onto the ROM basis after each time step [77], known as the re-projection approach. The resulting FOM snapshot data does not contain non-Markovian information and is equivalent to snapshot data generated by the intrusive ROM. Consequently, the minimization problem is unperturbed and the intrusive ROM does minimize it. However, operator inference is only guaranteed to yield this solution if this minimizer is unique. Concretely, the operator inference minimization problem is a least squares problem and has a unique solution only if the least squares matrix has full rank.

This full rank requirement is the second major difficulty in accurately reconstructing the intrusive ROM operators. To the authors' knowledge, no method has been proposed yet to ensure this full rank requirement a priori. In practice, two different approaches have been suggested to address this issue. The first approach is to generate large numbers of

To infer operators from data, the data must be sufficiently informative.

snapshots and test a posteriori whether the resulting least squares problem has full rank [77, 78]. However, this approach does not guarantee full rank, even when the number of generated snapshots exceeds the required rank by orders of magnitude, and hence causes large computational costs. The second approach is to circumvent the need for full rank by adding a regularization term to the operator inference least squares problem [101, 106]. However, this modification inevitably causes the resulting inferred ROM operators to deviate from the intrusive ROM operators.

In this chapter, we propose a novel snapshot data generation method that guarantees the full rank of the operator inference least squares problem. The number of generated snapshots is equal to this rank and hence minimal. Like Peherstorfer’s approach [77], our method avoids non-Markovian information and hence reconstructs the intrusive ROM operators exactly, i.e., up to machine precision. Our method has two key ingredients. First, we leverage the knowledge about the specific polynomial structure of the intrusive ROM to identify a set of ROM states and input signals that define snapshots that induce a full rank least squares problem. Second, we generate snapshots corresponding to these identified ROM states and input signals by simulating multiple trajectories, initialized with the identified ROM states as initial conditions and the identified input signals, for only a single explicit Euler time step. In contrast to the usual approach that generates snapshot data along long trajectories, this single-step approach gives us control over which snapshot data we obtain. Additionally, this single-step approach avoids any non-Markovian terms, so we do not need re-projection[77], and the explicit Euler scheme prevents time discretization inconsistencies. Since the number of snapshots is minimal, the least squares problem simplifies to a linear system that is numerically more stable. Since the method exactly reconstructs intrusive ROM operators, it preserves structures such as symmetry and skew-symmetry of these intrusive operators [98] without the need to enforce this preservation explicitly [45, 49, 62].

This chapter is organized as follows. In Section 3.2, we introduce dynamical systems with polynomial terms, intrusive projection-based model reduction, operator inference and the limitation of insufficient snapshot data. In Section 3.3, we propose our novel method for rank-sufficient snapshot data generation. In Section 3.4, we demonstrate the novel method in numerical experiments. In Section 3.5, we draw conclusions and give an outlook on future work.

3.2 PRELIMINARIES

Our notation is based on [77], but we start at the time-continuous level.

3.2.1 Dynamical systems with polynomial terms

As full-order model (FOM), we consider a time-continuous dynamical system

$$\dot{x}(t; \mu) = f(x(t; \mu), u(t; \mu); \mu), \quad x(0; \mu) = x_0(\mu), \quad (3.1)$$

with state $x(t; \mu) \in \mathbb{R}^N$, input $u(t; \mu) \in \mathbb{R}^{N_u}$, time $t \in [0, T]$, and parameter $\mu \in \mathcal{D} \subset \mathbb{R}^d$. We assume the nonlinear function $f : \mathbb{R}^N \times \mathbb{R}^{N_u} \times \mathcal{D} \rightarrow \mathbb{R}^N$ to be polynomial of order $l \in \mathbb{N}_0$ in the state x and linear in the input signal u , so there exist $\mathbf{A}_i(\mu) \in \mathbb{R}^{N \times N_i}$ with

$$N_i = \binom{N+i-1}{i} \text{ for } i \in \mathcal{I}, \text{ and } \mathbf{B}(\mu) \in \mathbb{R}^{N \times N_u} \text{ for } \mu \in \mathcal{D} \text{ such that}$$

$$f(x, u; \mu) = \sum_{i \in \mathcal{I}} \mathbf{A}_i(\mu) x^i + \mathbf{B}(\mu) u, \quad (3.2)$$

with the degree set $\mathcal{I} \subset \{0, 1, \dots, l\}$. The vector $x^i \in \mathbb{R}^{N_i}$ comprises all unique products of i entries of x (x^0 is scalar 1). We can construct this vector from the i -fold Kronecker product $x \otimes \dots \otimes x =: x^{i \otimes}$ by removing all duplicate entries due to commutativity of the multiplication [78]. Formally, we can write $x^i = \mathbf{I}_N^{(i)} x^{i \otimes}$ with $\mathbf{I}_N^{(i)} \in \{0, 1\}^{N_i \times N^i}$ an identity matrix with certain rows removed, and conversely $x^{i \otimes} = \mathbf{J}_N^{(i)} x^i$ with $\mathbf{J}_N^{(i)} \in \{0, 1\}^{N^i \times N_i}$ an identity matrix with certain rows copied and permuted. Here N^i denotes as usual the i -th power of N .

3.2.2 Intrusive projection-based model reduction

To obtain a projection-based reduced order model (ROM), a common intrusive approach is to project the FOM (3.1) onto $\mathbf{V}_n$ and replace the FOM state $x(t; \mu)$ by the lower-dimensional approximation $\mathbf{V}_n \tilde{x}(t; \mu)$,

$$\dot{\tilde{x}}(t; \mu) = \mathbf{V}_n^T f(\mathbf{V}_n \tilde{x}(t; \mu), u(t; \mu); \mu), \quad \tilde{x}(0; \mu) = \mathbf{V}_n^T x_0(\mu). \quad (3.3)$$

Here, $\mathbf{V}_n = [v_1, \dots, v_n] \in \mathbb{R}^{N \times n}$ is an orthonormal basis of an n -dimensional subspace of $\mathbb{R}^N$ with $n \ll N$, and $\tilde{x}(t; \mu) \in \mathbb{R}^n$ the ROM state. The orthonormal basis $\mathbf{V}_n$ can, for example, be obtained from proper orthogonal decomposition (POD) [57] of snapshots $x_k(\mu) \in \mathbb{R}^N$, $k = 0, \dots, K_{\text{POD}} - 1$ of a time discretization of the FOM (3.1). Other methods to construct reduced bases and details on how to sample parameters and the ROM dimension n are discussed in [12].

To allow efficient simulations of the ROM (3.3), the cost of evaluating the right-hand side must be independent of the FOM dimension N .

For this purpose, the intrusive approach is to precompute reduced operators,

$$\tilde{\mathbf{B}}(\boldsymbol{\mu}) := \mathbf{V}_n^T \mathbf{B}(\boldsymbol{\mu}) \in \mathbb{R}^{n \times N_u}, \quad (3.4)$$

and

$$\tilde{\mathbf{A}}_i(\boldsymbol{\mu}) := \mathbf{V}_n^T \mathbf{A}_i(\boldsymbol{\mu}) \mathbf{I}_N^{(i)} \mathbf{V}_n^{i \otimes} \mathbf{J}_N^{(i)} \in \mathbb{R}^{n \times n_i}, \quad (3.5)$$

with $n_i = \binom{n+i-1}{i}$ for all $i \in \mathcal{I}$, whose dimensions only depend on the ROM dimension n (and the unreduced input signal dimension N_u) [77]. Note that these precomputations must in general be performed for each choice of the parameter $\boldsymbol{\mu} \in \mathcal{D}$ separately. The resulting efficient ROM for fixed $\boldsymbol{\mu}$ is given by

$$\begin{aligned} \dot{\tilde{\mathbf{x}}}(t; \boldsymbol{\mu}) &= \tilde{\mathbf{f}}(\tilde{\mathbf{x}}(t; \boldsymbol{\mu}), \mathbf{u}(t; \boldsymbol{\mu}); \boldsymbol{\mu}) \\ &= \sum_{i \in \mathcal{I}} \tilde{\mathbf{A}}_i(\boldsymbol{\mu}) \tilde{\mathbf{x}}^i(t; \boldsymbol{\mu}) + \tilde{\mathbf{B}}(\boldsymbol{\mu}) \mathbf{u}(t; \boldsymbol{\mu}) \\ &= \tilde{\mathbf{O}}(\boldsymbol{\mu}) \mathbf{p}(\tilde{\mathbf{x}}(t; \boldsymbol{\mu}), \mathbf{u}(t; \boldsymbol{\mu})), \end{aligned} \quad (3.6)$$

with the aggregated ROM operator

$$\tilde{\mathbf{O}}(\boldsymbol{\mu}) := \left[[\tilde{\mathbf{A}}_i(\boldsymbol{\mu})]_{i \in \mathcal{I}} \quad \tilde{\mathbf{B}}(\boldsymbol{\mu}) \right] \in \mathbb{R}^{n \times n_f},$$

with $n_f = n_p + N_u$, $n_p = \sum_{i \in \mathcal{I}} n_i$ and the aggregated ROM state (and input) polynomial

$$\mathbf{p} : \mathbb{R}^n \times \mathbb{R}^{N_u} \rightarrow \mathbb{R}^{n_f}, \quad \mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} [\tilde{\mathbf{x}}^i]_{i \in \mathcal{I}} \\ \mathbf{u} \end{bmatrix}. \quad (3.7)$$

3.2.3 Operator inference

As described in the previous subsection, intrusive model reduction requires access to the FOM operators $\mathbf{A}_i(\boldsymbol{\mu})$, $\mathbf{B}(\boldsymbol{\mu})$ to precompute the ROM operators $\tilde{\mathbf{A}}_i(\boldsymbol{\mu})$, $\tilde{\mathbf{B}}(\boldsymbol{\mu})$ via (3.4) and (3.5). In practice, this access is not possible in many FOM solvers. Instead, operator inference aims to learn approximations $\hat{\mathbf{A}}_i(\boldsymbol{\mu}) \in \mathbb{R}^{n \times n_i}$, $\hat{\mathbf{B}}(\boldsymbol{\mu}) \in \mathbb{R}^{n \times N_u}$ of the intrusive ROM operators from data [78]. This data consists of triples of a ROM state snapshot $\check{\mathbf{x}}_k(\boldsymbol{\mu}) \in \mathbb{R}^n$, a ROM state time derivative snapshot $\dot{\check{\mathbf{x}}}_k(\boldsymbol{\mu}) \in \mathbb{R}^n$ and an input signal snapshot $\mathbf{u}_k(\boldsymbol{\mu})$, $k = 0, \dots, K-1$. For these triples, operator inference minimizes the residual of the ROM ODE (3.6) to infer the aggregated ROM operator,

$$\hat{\mathbf{O}}(\boldsymbol{\mu}) = [\hat{\mathbf{A}}_i]_{i \in \mathcal{I}} \quad \hat{\mathbf{B}}(\boldsymbol{\mu}) \quad \sum_{k=0}^{K-1} \|\hat{\mathbf{O}}(\boldsymbol{\mu}) \mathbf{p}(\check{\mathbf{x}}_k(\boldsymbol{\mu}), \mathbf{u}_k(\boldsymbol{\mu})) - \dot{\check{\mathbf{x}}}_k(\boldsymbol{\mu})\|_2^2. \quad (3.8)$$

If snapshot triples are obtained from the intrusive ROM ODE (3.6) such that $\check{x}_k(\boldsymbol{\mu}) = \tilde{x}(t_k; \boldsymbol{\mu})$, $\dot{\check{x}}_k(\boldsymbol{\mu}) = \dot{\tilde{x}}(t_k; \boldsymbol{\mu})$ and $\mathbf{u}_k(\boldsymbol{\mu}) = \mathbf{u}(t_k; \boldsymbol{\mu})$ at time steps t_k , $k = 0, \dots, K-1$, then the intrusive aggregated ROM operator $\tilde{\mathbf{O}}(\boldsymbol{\mu}) = [[\tilde{\mathbf{A}}_i]_{i \in \mathcal{I}} \quad \tilde{\mathbf{B}}(\boldsymbol{\mu})]$ is a solution to this minimization problem.

In practice, however, snapshots of the ROM ODE (3.6) are usually not available, but only snapshots of time discretizations of the FOM (3.1), $\mathbf{x}_k(\boldsymbol{\mu}) \in \mathbb{R}^N$, $\mathbf{u}_k(\boldsymbol{\mu}) \in \mathbb{R}^{N_u}$ corresponding to time steps t_k , $k = 0, \dots, K$. These snapshots can be used to approximate the ROM snapshots $\check{x}_k(\boldsymbol{\mu})$ in (3.8) by projecting onto the ROM basis,

$$\check{x}_k(\boldsymbol{\mu}) = \mathbf{V}_n^T \mathbf{x}_k(\boldsymbol{\mu}), \quad k = 0, \dots, K, \quad (3.9)$$

and to approximate the time derivative snapshots $\dot{\check{x}}_k(\boldsymbol{\mu})$ via some finite difference, in the simplest case explicit Euler,

$$\dot{\check{x}}_k(\boldsymbol{\mu}) = \frac{\check{x}_{k+1}(\boldsymbol{\mu}) - \check{x}_k(\boldsymbol{\mu})}{t_{k+1} - t_k}, \quad k = 0, \dots, K-1. \quad (3.10)$$

When snapshots (3.9) and (3.10) are used in the operator inference minimization problem (3.8), the inferred operator $\hat{\mathbf{O}}(\boldsymbol{\mu})$ resulting from solving this minimization problem is generally not equal to the intrusive operator $\tilde{\mathbf{O}}(\boldsymbol{\mu})$, but only an approximation. This approach is non-intrusive because it does not require access to the FOM operators $\tilde{\mathbf{A}}_i(\boldsymbol{\mu})$, $\tilde{\mathbf{B}}(\boldsymbol{\mu})$, but only the knowledge about $\mathcal{I}$ and N_u , and FOM snapshot data.

Note that operator inference for ROMs with POD bases take two sets of snapshots as input. The first set, the POD snapshot data, consists of FOM state snapshots $\mathbf{x}_k(\boldsymbol{\mu}) \in \mathbb{R}^N$, $k = 0, \dots, K_{\text{POD}} - 1$ that are used to compute the POD basis $\mathbf{V}_n$, exactly in the same way as for the corresponding intrusive ROM with POD basis. The second set, the operator inference snapshot data, consists of triples of a ROM state snapshot $\check{x}_k \in \mathbb{R}^n$, a ROM state time derivative snapshot $\dot{\check{x}}_k \in \mathbb{R}^n$ and an input signal snapshot $\mathbf{u}_k \in \mathbb{R}^{N_u}$. This set defines the minimization problem (3.8).

Note further that in general the inference of the ROM operators by solving (3.8) must be performed separately for each choice of the parameter $\boldsymbol{\mu}$, just like the intrusive operator precomputations (3.4) and (3.5). For parameter values for which inference data is not available, the corresponding operators are then obtained by interpolation of inferred operators [78]. Under certain assumptions, the parameter dependency of the ROM operators can explicitly be incorporated in the operator inference minimization problem (3.8) [70]. In this chapter, we consider the parameter $\boldsymbol{\mu}$ fixed and will omit it in the following sections.

3.2.4 Insufficiently informative snapshot data

As has been observed by several researchers [45, 69, 77], standard operator inference via (3.8) often fails to accurately reproduce the intrusive ROM operators. There are several causes of such failures such as inconsistency between the FOM time discretization and the discretization of the time derivative snapshots $\dot{\mathbf{x}}_k(\boldsymbol{\mu})$ or non-Markovian information in the snapshot data [77]. In this chapter, we want to highlight the remaining cause that has not been resolved yet: insufficiently informative snapshot data. For this purpose, we write (3.8) equivalently as

$$\min_{\hat{\mathbf{O}} \in \mathbb{R}^{n \times n_f}} \left\| \hat{\mathbf{O}} \mathbf{P} - \hat{\mathbf{X}} \right\|_F^2, \quad (3.11)$$

$$\text{with } \mathbf{P} := [\mathbf{p}(\check{\mathbf{x}}_0, \mathbf{u}_0) \ \dots \ \mathbf{p}(\check{\mathbf{x}}_{K-1}, \mathbf{u}_{K-1})] \in \mathbb{R}^{n_f \times K} \quad (3.12)$$

$$\text{and } \hat{\mathbf{X}} := [\dot{\check{\mathbf{x}}}_0 \ \dots \ \dot{\check{\mathbf{x}}}_{K-1}] \in \mathbb{R}^{n \times K}. \quad (3.13)$$

This optimization problem has a unique solution $\hat{\mathbf{O}}$ if and only if the matrix $\mathbf{P}$ has full rank n_f [77]. To compute this solution numerically, also the numerical rank of $\mathbf{P}$ — defined as the number of singular values larger than a specified tolerance — must be n_f . In general, there is no guarantee that this requirement is satisfied, even if the number of snapshots K is much larger than n_f . As described in [77], strategies to mitigate such rank deficiency include regularization and reduction of the ROM dimension n . However, these strategies do not guarantee accurate reconstruction of the intrusive ROM operators. Regularization can limit the accuracy of the resulting operators. Indeed, many works on regularized operator inference report significant differences between the non-intrusive ROMs and their intrusive counterparts [11, 70, 101]. Reducing the ROM dimension, on the other hand, can result in a least squares matrix of even smaller rank than the initial matrix. Therefore, iterative reduction of the ROM dimension can result in a significantly smaller dimension n corresponding to an intrusive ROM of insufficient accuracy.

In the numerical experiments in [77], full numerical rank has been achieved by generating and concatenating multiple trajectories of snapshots with randomized input signals. However, the number and length of the trajectories and the distributions of the random input signals are hyperparameters that have to be tuned, and whether the resulting least squares matrix has full numerical rank can only be checked a posteriori.

Without sufficiently informative snapshot data, the operator inference problem solution is not uniquely defined.

3.3 EXACT OPERATOR INFERENCE

In this chapter, we propose a novel method to generate snapshots that is a priori guaranteed to result in a full rank matrix $\mathbf{P}$. Our method has two key ingredients. First, in Section 3.3.1, we leverage the knowledge about how the matrix $\mathbf{P}$ depends on the ROM states $\tilde{\mathbf{x}}_k$ and input signals $\mathbf{u}_k$ via (3.12) and (3.7) to find pairs of ROM states $\hat{\mathbf{x}}_s \in \mathbb{R}^n$ and input signals $\bar{\mathbf{u}}_s \in \mathbb{R}^{N_u}$, $s = 1, \dots, n_f$ that ensure full rank of $\mathbf{P}$. Second, in Section 3.3.2, we generate time derivative snapshots $\hat{\mathbf{x}}_s$ corresponding to these rank-ensuring pairs $(\hat{\mathbf{x}}_s, \bar{\mathbf{u}}_s)$ by simulating n_f trajectories for only a single time step each, with initial conditions $\hat{\mathbf{x}}_s$ and input signals $\bar{\mathbf{u}}_s$.

3.3.1 Rank-ensuring ROM states and input signals

The motivation for our choice of rank-ensuring ROM states is to isolate single entries of the aggregated polynomial $\mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u})$ defined in (3.7). If we could find inputs $\tilde{\mathbf{x}}$ and $\mathbf{u}$ such that $\mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u})$ is 1 in the j -th entry and zero in all other entries for all $j = 1, \dots, n_f$, then the product $\hat{\mathbf{O}}\mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u})$ would equal the j -th column of the operator $\hat{\mathbf{O}}$. Hence, we could easily infer all columns of the operator $\hat{\mathbf{O}}$.

Example 1. If $\mathcal{I} = \{1\}$ and $N_u = 0$, so $\mathbf{p}(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}$, we can achieve such an isolation of entries by using the unit vectors in $\mathbb{R}^n$, here denoted by $\mathbf{e}_j$, $j = 1, \dots, n$, as inputs $\tilde{\mathbf{x}}$.

Example 2. If $\mathcal{I} = \{2\}$ and $N_u = 0$, so $\mathbf{p}(\tilde{\mathbf{x}}) = \tilde{\mathbf{x}}^2$, perfect isolation of entries is not possible. For example, when choosing $\tilde{\mathbf{x}} = [\tilde{x}_1, \dots, \tilde{x}_n]^T = \mathbf{e}_i + \mathbf{e}_j$ such that $\mathbf{p}(\tilde{\mathbf{x}})$ is 1 in the entry $\tilde{x}_i\tilde{x}_j$ with $i \neq j$, also the entries $\tilde{x}_i^2$ and $\tilde{x}_j^2$ are 1. However, the columns of $\hat{\mathbf{O}}$ corresponding to $\tilde{x}_i\tilde{x}_j$, $\tilde{x}_i^2$ and $\tilde{x}_j^2$ can still be inferred when employing the data $\hat{\mathbf{O}}\mathbf{p}(\mathbf{e}_i + \mathbf{e}_j)$, $\hat{\mathbf{O}}\mathbf{p}(2\mathbf{e}_i)$ and $\hat{\mathbf{O}}\mathbf{p}(2\mathbf{e}_j)$.

Based on this motivation, we define for arbitrary $i \in \mathbb{N}^+$ the set $\mathcal{X}^i$ that consists of all sums of i of these vectors,

$$\mathcal{X}^i := \left\{ \sum_{j=1}^i \mathbf{a}_j \mid \mathbf{a}_1, \dots, \mathbf{a}_i \in \{\mathbf{e}_1, \dots, \mathbf{e}_n\} \right\}$$

and consistently $\mathcal{X}^0 := \{\mathbf{0} \in \mathbb{R}^n\}$. (3.14)

The main result of this chapter is: the vectors in $\cup_{i \in \mathcal{I}} \mathcal{X}^i$ induce full rank of the matrix $\mathbf{P}$ as defined in (3.12) and (3.7), for any degree set $\mathcal{I} \subset \mathbb{N}_0$ and $N_u = 0$.

If $N_u > 0$, we achieve full rank of $\mathbf{P}$ by pairing these ROM states $\hat{\mathbf{x}}_j$ with zero input signals, $(\hat{\mathbf{x}}_j, \mathbf{0})$, and adding the pairs $(\mathbf{0}, \mathbf{u}_j)$ with zero

ROM state and $\mathbf{u}_j$, $j = 1, \dots, N_u$ the unit vectors in $\mathbb{R}^{N_u}$. This way we get the rank-ensuring snapshot pairs:

$$\{(\hat{\mathbf{x}}_j, \bar{\mathbf{u}}_j)\}_{j=1}^{n_f} = \{(\hat{\mathbf{x}}_j, \mathbf{0}) | \tilde{\mathbf{x}}_j \in \cup_{i \in \mathcal{I}} \mathcal{X}^i\} \cup \{(\mathbf{0}, \mathbf{u}_j)\}_{j=1}^{N_u}. \quad (3.15)$$

Example 3. For $n = 2$, $N_u = 2$ and $\mathcal{I} = \{1, 2\}$ (such that $n_p = 5$, $n_f = 7$), we get the pairs

$$\underbrace{\left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right), \left(\begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right)}_{\{(\hat{\mathbf{x}}_j, \mathbf{0}) | \tilde{\mathbf{x}}_j \in \mathcal{X}^1\}},$$

$$\underbrace{\left(\begin{bmatrix} 2 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right), \left(\begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right), \left(\begin{bmatrix} 0 \\ 2 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right)}_{\{(\hat{\mathbf{x}}_j, \mathbf{0}) | \tilde{\mathbf{x}}_j \in \mathcal{X}^2\}},$$

$$\underbrace{\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right), \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix} \right)}_{\{(\mathbf{0}, \mathbf{u}_j)\}_{j=1}^{N_u}},$$

that result in the full-rank matrix

$$\mathbf{P} = \begin{bmatrix} 1 & 0 & 2 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 2 & 0 & 0 \\ 1 & 0 & 4 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{where} \quad \mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \tilde{x}_1^2 \\ \tilde{x}_1 \tilde{x}_2 \\ \tilde{x}_2^2 \\ u_1 \\ u_2 \end{bmatrix}.$$

In general, the following theorem holds.

Theorem 3.1 (Full rank of matrix $\mathbf{P}$). For any $n \in \mathbb{N}^+$, $N_u \in \mathbb{N}_0$ and $\mathcal{I} \subset \mathbb{N}_0$, the matrix $\mathbf{P} := [\mathbf{p}(\tilde{\mathbf{x}}_0, \mathbf{u}_0) \dots \mathbf{p}(\tilde{\mathbf{x}}_{K-1}, \mathbf{u}_{K-1})] \in \mathbb{R}^{n_f \times K}$ defined

in (3.12) with $K = n_f$, $\mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u}) = \begin{bmatrix} \tilde{\mathbf{x}}^i \\ \mathbf{u} \end{bmatrix}_{i \in \mathcal{I}}$ as defined in (3.7) and with

the ROM states and input signals $\{(\hat{\mathbf{x}}_j, \bar{\mathbf{u}}_j)\}_{j=1}^{n_f} = \{(\hat{\mathbf{x}}_j, \mathbf{0}) | \tilde{\mathbf{x}}_j \in \cup_{i \in \mathcal{I}} \mathcal{X}^i\} \cup \{(\mathbf{0}, \mathbf{u}_j)\}_{j=1}^{N_u}$ defined in (3.15) has full rank.

To find rank-ensuring snapshots, we interpret $\mathbf{P}$ as interpolation matrix and identify interpolation nodes that uniquely define the interpolant (see Appendix B.1).

The proof is given in Appendix B.1. For moderate values of n , N_u and $\mathcal{I}$ which are relevant in practice, the rank and the numerical rank of $\mathbf{P}$ coincide, so we will only refer to the rank in the following.

Note that this choice of ROM states and input signals results in a matrix $\mathbf{P}$ that has as many columns as rows. This means that our choice uses the minimal number of snapshots necessary to achieve full rank. In addition, because $\mathbf{P}$ is square, the operator inference minimization problem (3.11) is equivalent to the matrix-valued linear system

$$\hat{\mathbf{O}}\mathbf{P} = \hat{\mathbf{X}}. \quad (3.16)$$

Since the matrix $\mathbf{P}$ has a smaller condition number than the matrix $\mathbf{P}\mathbf{P}^T$ of the normal equations

$$\hat{\mathbf{O}}\mathbf{P}\mathbf{P}^T = \hat{\mathbf{X}}\mathbf{P}^T \quad (3.17)$$

corresponding to the least squares problem (3.11), the linear system (3.16) is in general less sensitive to error amplifications.

3.3.2 Snapshot data generation via single-step ensemble

To formulate the operator inference minimization problem (3.8), we not only need the pairs of ROM states and input signals, denoted by $(\hat{\mathbf{x}}_j, \bar{\mathbf{u}}_j)$, $j = 1, \dots, n_f$ defined in (3.15), but also approximations of the ROM state time derivative $\hat{\dot{\mathbf{x}}}_j$ that correspond to these pairs. In existing works on operator inference [45, 69, 77, 78], triples of these quantities are obtained by projection of the corresponding FOM vector that are generated along trajectories of the FOM. Hence, the FOM state $\mathbf{x}_j$ at a given time step t_j depends on the FOM state $\mathbf{x}_{j-1}$ at the previous time step t_{j-1} and the FOM dynamics (3.2). These FOM dynamics are unknown (otherwise we would not need to perform operator inference). Consequently, we cannot predict or control which FOM states are attained along such trajectories. In particular, we cannot guarantee that FOM states corresponding to the ROM states $\hat{\mathbf{x}}_j$ described in the previous section are attained.

In our approach, we do not generate FOM snapshot data along one (or few) long trajectories but instead use an ensemble of many short trajectories, each containing *only a single time step*. Each trajectory is initialized with

$$\begin{aligned} &\text{the rank-ensuring initial condition } \mathbf{x}_0^s = \mathbf{V}_n \hat{\mathbf{x}}_s, \\ &\text{and input signal } \mathbf{u}_0^s = \bar{\mathbf{u}}_s, \quad s = 1, \dots, n_f, \end{aligned} \quad (3.18)$$

with $\hat{\mathbf{x}}_s$ and $\bar{\mathbf{u}}_s$ as defined in (3.15), and integrated in time for only a single explicit Euler step of size Δt . This approach results in FOM snapshots

$$\mathbf{x}_1^s = \mathbf{x}_0^s + \Delta t \mathbf{f}(\mathbf{x}_0^s, \mathbf{u}_0^s) \quad s = 1, \dots, n_f. \quad (3.19)$$

*Along each trajectory,
we can only control the
initial condition.*

*Prescribe all
rank-ensuring ROM
states as initial
conditions so that the
corresponding snapshot
data is generated.*

From each pair of FOM snapshots $\mathbf{x}_0^s$ and $\mathbf{x}_1^s$, we can compute the ROM time derivative snapshot

$$\hat{\mathbf{x}}_s = \mathbf{V}_n^T \frac{\mathbf{x}_1^s - \mathbf{x}_0^s}{\Delta t} \quad s = 1, \dots, n_f. \quad (3.20)$$

The resulting triples $(\hat{\mathbf{x}}_s, \hat{\mathbf{x}}_s, \mathbf{u}_s)_{s=1}^{n_f}$ form the input to the operator inference optimization problem (3.8). Since the matrix $\mathbf{P}$ (3.12) only depends on the snapshots $\hat{\mathbf{x}}_s$ and $\mathbf{u}_s$, $s = 1, \dots, n_f$, it does not depend on the unknown FOM operators. Hence, we can guarantee full rank of $\mathbf{P}$ by the choice of the initial conditions and input signals.

Note that our novel approach for generating snapshot data for operator inference does not affect the snapshot data used for POD, so the POD basis $\mathbf{V}_n$ remains unchanged. For example, the implicit Euler method is used to generate POD snapshot data in Section 3.4.2.

The idea to generate snapshot data in multiple trajectories, each containing only a single time step, has already been used in numerical experiments in [78]. However, randomized instead of well-chosen initial conditions are used in that work. Later, FOM snapshots projected onto the ROM space have been used as initial conditions in context of lifted nonlinearities [86] and noisy data [110]. All three approaches enable to check the full rank a priori, but do not guarantee full rank. Indeed, far more than n_f trajectories have been used to achieve full rank of $\mathbf{P}$ in the numerical experiments in these works.

The complete method is summarized in Algorithm 3.1.

3.3.3 Exact reconstruction of the intrusive ROM operators

By combining the rank-ensuring initial conditions and input signals in Section 3.3.1 and the single-step ensemble in Section 3.3.2, our novel approach exactly reconstructs the intrusive ROM operators, as stated by the following theorem.

Theorem 3.2 (Exact reconstruction of intrusive ROM operators). *For any $n \in \mathbb{N}^+$, $N_u \in \mathbb{N}_0$, $\mathcal{I} \subset \mathbb{N}_0$ and $\Delta t > 0$, the operator inference minimization problem (3.8) with snapshot data $(\check{\mathbf{x}}_k, \check{\mathbf{x}}_k, \mathbf{u}_k) = (\hat{\mathbf{x}}_s, \hat{\mathbf{x}}_s, \mathbf{u}_s)$ defined in (3.15) and (3.20) has a unique solution. This solution is equivalent to the intrusive ROM operators defined in (3.4) and (3.5).*

Proof. We first show that the FOM snapshot data generated in our proposed approach is equivalent to snapshot data generated from the intrusive ROM itself, which constitutes the ideal inference data. For this purpose, we insert the definition of the rank-ensuring initial conditions

Algorithm 3.1 Exact operator inference

```

1: procedure EXACTOPINF(
    ROM basis  $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ ,
    degree set  $\mathcal{I} \subset \mathbb{N}_0$ ,
    input signal dimension  $N_u \in \mathbb{N}_0$ 
    time step size  $\Delta t$ 
)
2:   # Generate rank-ensuring snapshot data.
3:   Initialize  $\mathbf{P} \leftarrow \emptyset$ 
4:   Initialize  $\hat{\mathbf{X}} \leftarrow \emptyset$ 
5:   for  $i \in \mathcal{I}$  do
6:       # Generate ROM state sets  $\mathcal{X}^i$  as defined in (3.14)
       (see Appendix B.2)
7:        $\mathcal{X}^i \leftarrow \text{GENERATEROMSTATESET}(\text{ROM dimension } n, \text{polyno-}$ 
       mial degree  $i$ )
8:   end for
9:   for  $\hat{x} \in \cup_{i \in \mathcal{I}} \mathcal{X}^i$  do
10:       $x_1 \leftarrow \text{FOM}(\text{initial condition } x_0 = \mathbf{V}_n \hat{x},$ 
        input signal  $u = \mathbf{0}$ , time step size  $\Delta t$ ,
        end time  $T = \Delta t$ ,
        explicit Euler time discretization )
11:       $\mathbf{P} \leftarrow [\mathbf{P}, p(\tilde{x}_0, \mathbf{0})]$ 
12:       $\hat{\mathbf{X}} \leftarrow [\hat{\mathbf{X}}, \frac{\mathbf{V}_n^T x_1 - \hat{x}}{\Delta t}]$ 
13:   end for
14:   for  $j = 1, \dots, N_u$  do
15:       $u \leftarrow j\text{-th unit vector in } \mathbb{R}^{N_u}$ 
16:       $x_1 \leftarrow \text{FOM}(\text{initial condition } x_0 = \mathbf{0},$ 
        input signal  $u$ , time step size  $\Delta t$ ,
        end time  $T = \Delta t$ ,
        explicit Euler time discretization )
17:       $\mathbf{P} \leftarrow [\mathbf{P}, p(\mathbf{0}, u)]$ 
18:       $\hat{\mathbf{X}} \leftarrow [\hat{\mathbf{X}}, \frac{\mathbf{V}_n^T x_1 - \mathbf{0}}{\Delta t}]$ 
19:   end for
20:   Solve linear system (3.16)
21:   return  $\hat{\mathbf{O}}$ 
22: end procedure

```

$\hat{x}_s$ (3.18) and the explicit Euler FOM time step (3.19) into the definition of the explicit Euler time derivative snapshot (3.20) to find

$$\begin{aligned}\hat{x}_s &= \frac{1}{\Delta t} \mathbf{V}_n^T (\mathbf{x}_0^s + \Delta t \mathbf{f}(\mathbf{x}_0^s, \mathbf{u}_0^s) - \mathbf{x}_0^s) = \mathbf{V}_n^T \mathbf{f}(\mathbf{x}_0^s, \mathbf{u}_0^s) \\ &= \mathbf{V}_n^T \mathbf{f}(\mathbf{V}_n \hat{x}_s, \mathbf{u}_0^s) = \tilde{\mathbf{O}} \mathbf{p}(\hat{x}_s, \mathbf{u}_0^s).\end{aligned}\quad (3.21)$$

Hence, (3.8) simplifies to

$$\min_{\hat{\mathbf{O}}} \sum_{s=1}^{n_f} \|\hat{\mathbf{O}} \mathbf{p}(\hat{x}_s, \mathbf{u}_0^s) - \tilde{\mathbf{O}} \mathbf{p}(\hat{x}_s, \mathbf{u}_0^s)\|_2^2, \quad (3.22)$$

or equivalently

$$\min_{\hat{\mathbf{O}}} \|\hat{\mathbf{O}} \mathbf{P} - \tilde{\mathbf{O}} \mathbf{P}\|_F^2. \quad (3.23)$$

Since $\mathbf{P}$ has full rank as shown in Theorem 3.1, the solution $\hat{\mathbf{O}} = \tilde{\mathbf{O}}$ is unique, so the method exactly reconstructs the intrusive ROM operator. $\square$

Note that our approach not only guarantees the full rank of $\mathbf{P}$ but also eliminates two other error sources common in operator inference. Firstly, the consistency of using explicit Euler schemes, both for the snapshot generation (3.19) and the computation of the time derivative snapshots (3.20) avoids the introduction of time discretization inconsistencies. Secondly, by performing only a single explicit Euler time step on each trajectory we avoid introducing non-Markovian terms that would cause the projected FOM snapshot data (3.9) to deviate from actual ROM snapshot data as described in [77].

Since our approach exactly reconstructs intrusive ROM operators, it also preserves any structure present in these intrusive operators such as symmetry and skew-symmetry [98]. In contrast to other works [45, 49, 62], these structures are preserved automatically without the need to enforce their preservation explicitly.

3.3.4 Assumptions on the FOM solver

Operator inference is a non-intrusive model reduction concept developed for FOM solvers that do not provide the access to FOM operators required by intrusive model reduction. Consequently, imposing strong assumptions on the FOM solver contradicts the core motivation for operator inference. In our proposed approach, we only impose two assumptions.

First, we assume that specific initial conditions can be imposed; concretely, linear combinations of the ROM basis modes (3.18). For POD

*prescribe POD-based
initial conditions*

bases, these modes are linear combinations of FOM snapshots. Since these snapshots are physically realizable, the specific initial conditions are physically realizable as well in most cases. Exceptions include models that only allow a certain range of state values, for example positivity of temperature, or require specific boundary conditions. These exceptions are beyond the scope of this thesis since they typically require some additional treatment already on the intrusive ROM level, for example the introduction of a lifting function [90] to satisfy boundary conditions.

Second, we assume that explicit Euler can be chosen as time discretization method to generate the operator inference snapshot data. However, this restriction can be relaxed to higher-order Runge-Kutta schemes if input signals are interpolated between time steps. These schemes can be written as explicit Euler schemes whose right-hand side has higher polynomial degree [77]. Consequently, our approach can be applied to explicit Runge-Kutta methods (and multi-step methods that use an explicit Runge-Kutta method to compute the first time step). For implicit methods, on the other hand, the full rank of $\mathbf{P}$ might not be guaranteed. For example, if the FOM time steps (3.19) are replaced by implicit Euler steps,

$$\mathbf{x}_1^s = \mathbf{x}_0^s + \Delta t \mathbf{f}(\mathbf{x}_1^s, \mathbf{u}_1^s), \quad (3.24)$$

then the matrix $\mathbf{P}$ should, for time discretization consistency, be composed of vectors $\mathbf{p}(\mathbf{x}_1^s, \mathbf{u}_1^s)$ instead of $\mathbf{p}(\mathbf{x}_0^s, \mathbf{u}_0^s)$. However, in contrast to the initial conditions $\mathbf{x}_0^s$, we do not have full control over the states $\mathbf{x}_1^s$ after one time step. Consequently, we generally cannot guarantee the full rank of $\mathbf{P}$. Alternatively, we can construct the matrix $\mathbf{P}$ from the vectors $\mathbf{p}(\mathbf{x}_0^s, \mathbf{u}_0^s)$. This way, we guarantee the full rank of $\mathbf{P}$ but introduce a time discretization inconsistency that impedes exact reconstruction of the intrusive ROM operators.

Note that this requirement on the time discretization only affects the generation of operator inference snapshot data, but neither the generation of POD snapshot data, nor the simulations that are performed with the resulting ROM. The POD snapshot data can be generated from any FOM time discretization, and — like intrusive ROM operators — the inferred operators can be used for any ROM time discretization.

Note further that the time step size Δt is provided as an input in Algorithm 3.1 only for ease of presentation. Our proposed approach is also applicable to FOM solver that do not allow the users to specify the time step size themselves. In such cases, Δt is not provided as an input and the end time T cannot be chosen to be equal to Δt . Consequently, we can generally not guarantee that the FOM solver only performs a single time step, but by setting T to a small value, we can limit the number of excess time steps. From the solver output, we then only use

*use explicit time
discretization method*

the state snapshot $\mathbf{x}_1$ after the first time step and the corresponding time t_1 to compute the time step size Δt .

While our approach differs decisively from the re-projection approach [77] in terms of control over which snapshots are generated, both approaches are very similar in terms of intrusiveness. In fact, our approach can be seen as the re-projection approach performed for n_f separate trajectories, but stopped after a single time step and before the first re-projection step. Consequently, the described assumptions on the FOM solver are almost the same as for the re-projection approach. The only exception are implicit time discretization schemes for linear models which are allowed in re-projection but not in our approach because they inhibit control over the generated snapshots.

3.3.5 Remarks on changing ROM dimension

When ROMs are used in practice, it can be useful to change their dimension. For instance, the ROM dimension might be increased to improve accuracy or reduced to improve computational efficiency. The proposed data generation method is flexible towards such dimension changes. Concretely, the set $\mathbf{X}_n$ of FOM snapshots $\mathbf{x}_1^s$ as defined in (3.19) for any ROM dimension $n \in \mathbb{N}$ is a subset of the corresponding set $\mathbf{X}_{n+1}$ of FOM snapshots for dimension $n + 1$. Generally, $\mathbf{X}_{n_-} \subset \mathbf{X}_n \subset \mathbf{X}_{n_+}$ for all $n_- < n < n_+$. Consequently, we do not need to generate new data to infer ROM operators of dimension n_- when the snapshot data $\mathbf{X}_n$ is already generated, but can select the subset $\mathbf{X}_{n_-}$ of $\mathbf{X}_n$. Likewise, we do not need to generate the complete snapshot data $\mathbf{X}_{n_+}$ when $\mathbf{X}_n$ is given, but can completely reuse $\mathbf{X}_n$ and only have to generate $\mathbf{X}_{n_+} \setminus \mathbf{X}_n$ additionally.

Note that re-projection data [77] does not have this flexibility because the re-projection process is specific to a fixed ROM dimension.

3.3.6 Choice of time step size Δt

As stated in Theorem 3.2, our proposed method achieves exact reconstruction for any $\Delta t > 0$. Indeed, in exact arithmetic, the size of Δt also does not affect the accuracy of the inferred operators since the effect of Δt in computing $\mathbf{x}_1^s$ in (3.19) cancels out when computing the time derivative snapshot $\dot{\mathbf{x}}_1^s$ in (3.20). In floating point arithmetic, however, the accuracy of the inferred operators depends on how accurate the intermediate results can be represented as floating-point numbers. If Δt is determined by the FOM solver, we have to accept the inaccuracy in the ROM operator reconstruction induced by this limitations of floating-

point number representation. However, if Δt can be specified by the user, we can choose Δt such that these limitations are mitigated.

Concretely, the differences $\tilde{x}_1^s - \tilde{x}_0^s$ cannot be represented well as floating-point numbers if $\|\tilde{\mathbf{O}}\|_2$ is very large or very small. To counteract this effect, $\Delta t = \frac{1}{\|\tilde{\mathbf{O}}\|_2}$ would be a good choice. Since we do not know $\tilde{\mathbf{O}}$ a priori, we can only approximate $\|\tilde{\mathbf{O}}\|_2$ based on the POD snapshot data.

In this thesis, we use the estimate

$$\Delta t = \left(\max_{k=0, \dots, K_{\text{POD}}-1} \frac{\left\| \mathbf{v}_1^T \frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{t_{k+1} - t_k} \right\|_2}{\left\| \mathbf{p}(\mathbf{v}_1^T \mathbf{x}_k, \mathbf{u}(t_k)) \right\|_2} \right)^{-1}, \quad (3.25)$$

based on the lower bound

$$\begin{aligned} \|\tilde{\mathbf{O}}\|_2 &:= \sup_{\mathbf{y} \in \mathbb{R}^{n_f}} \frac{\|\tilde{\mathbf{O}}\mathbf{y}\|_2}{\|\mathbf{y}\|_2} \geq \sup_{\tilde{\mathbf{x}} \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^{N_u}} \frac{\|\tilde{\mathbf{O}}\mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u})\|_2}{\|\mathbf{p}(\tilde{\mathbf{x}}, \mathbf{u})\|_2} \\ &\geq \max_{k=0, \dots, K_{\text{POD}}-1} \frac{\|\tilde{\mathbf{O}}\mathbf{p}(\mathbf{V}_n^T \mathbf{x}_k, \mathbf{u}(t_k))\|_2}{\|\mathbf{p}(\mathbf{V}_n^T \mathbf{x}_k, \mathbf{u}(t_k))\|_2}, \end{aligned} \quad (3.26)$$

and the approximation

$$\tilde{\mathbf{O}}\mathbf{p}(\mathbf{V}_n^T \mathbf{x}_k, \mathbf{u}(t_k)) \approx \mathbf{V}_n^T \frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{t_{k+1} - t_k}, \quad k = 0, \dots, K_{\text{POD}} - 1, \quad (3.27)$$

for ROM dimension $n = 1$, so $\mathbf{V}_n = \mathbf{v}_1$.

Note that we do not need to take stability into account when determining the time step size Δt since we only perform a single time step per trajectory.

3.4 NUMERICAL EXPERIMENTS

The following numerical experiments demonstrate that Algorithm 3.1 exactly reconstructs the corresponding intrusive ROM operators. Many existing works on operator inference assess the accuracy of the inferred ROMs based on trajectory errors. As shown in [101], trajectory errors can be misleading because they are sensitive to the choice of trajectories. Furthermore, the objective of operator inference is to reconstruct the intrusive ROM, so this intrusive ROM should serve as ground truth, not the underlying FOM (see Appendix B.3). We therefore assess the reconstruction accuracy with relative operator errors $\frac{\|\hat{\mathbf{O}} - \tilde{\mathbf{O}}\|_F}{\|\tilde{\mathbf{O}}\|_F}$.

The code is available on <https://github.com/h3rr0r/exactOpInf>.

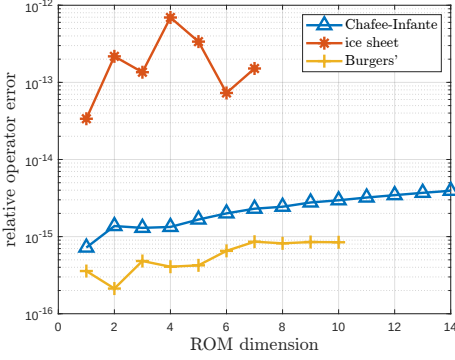


Figure 3.1: Relative operator errors $\frac{\|\hat{\mathbf{O}} - \tilde{\mathbf{O}}\|_F}{\|\tilde{\mathbf{O}}\|_F}$.

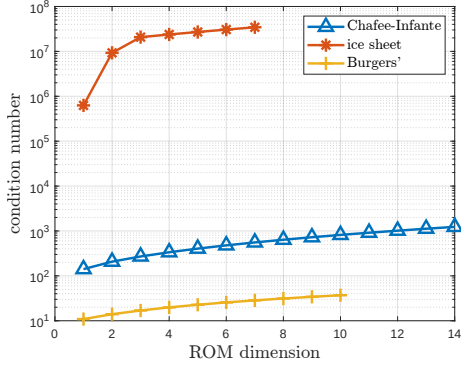


Figure 3.2: Condition numbers of data matrices $\mathbf{P}$.

3.4.1 Chafee-Infante equation / Allen-Cahn equation

We consider the Chafee-Infante equation / Allen-Cahn equation in a setup based on [77, 101],

$$\frac{\partial}{\partial t} x(\xi, t) = \frac{\partial^2}{\partial \xi^2} x(\xi, t) - x^3(\xi, t) + x(\xi, t), \quad (3.28)$$

with spatial coordinate $\xi \in \Omega = (0, 1)$ and time $t \in [0, 0.1]$. We impose the boundary conditions

$$x(0, t) = u(t), \quad \frac{\partial}{\partial \xi} x(1, t) = 0, \quad t \in [0, 0.1], \quad (3.29)$$

and zero initial condition $x(\xi, 0) = 0$ for $\xi \in \Omega$. We discretize with central finite differences on an equidistant grid in space with mesh width $\Delta \xi = 2^{-7}$, and with explicit Euler in time with time step size $\Delta t = 10^{-5}$, to obtain a time-discrete dynamical system of dimension $N = 128$,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t \left(\mathbf{A}_1 \mathbf{x}_k + \mathbf{A}_2 \mathbf{x}_k^2 + \mathbf{A}_3 \mathbf{x}_k^3 + \mathbf{B} u_k \right), \quad k = 0, \dots, K-1. \quad (3.30)$$

This system is polynomial of degree $l = 3$ with $\mathbf{A}_1 \in \mathbb{R}^{N \times N}$, $\mathbf{A}_2 \in \mathbb{R}^{N \times N_2}$ with $N_2 = \frac{N(N+1)}{2}$, $\mathbf{A}_3 \in \mathbb{R}^{N \times N_3}$ with $N_3 = \frac{N(N+1)(N+2)}{6}$, $\mathbf{B} \in \mathbb{R}^{N \times N_u}$ with $N_u = 1$, the input signal $u_k = 10(\sin(k\pi\Delta t) + 1)$, $k = 0, \dots, K-1$ and $K = 10^4$. The resulting $10^4 + 1$ snapshots are used to compute the POD bases $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ with $n = 1, \dots, 14$.

We apply Algorithm 3.1 with $\Delta t = 3.2741 \cdot 10^{-5}$ according to estimate (3.25) and compute the relative operator errors for ROM dimensions

$n = 1, \dots, 14$. Fig. 3.1 shows that the operator error is on the order of machine precision and Fig. 3.2 shows that the data matrices $\mathbf{P}$ have moderate condition numbers.

A crucial advantage of our method is the minimal number of FOM time steps needed for exact reconstruction of the intrusive ROM. For ROM dimension $n = 14$, we only need $n_f = n_1 + n_2 + n_3 + N_u = 680$ FOM time steps. This number is orders of magnitude smaller than in previous works with similar setups. For example, in [77], 10^7 FOM time steps with re-projection have been used to reconstruct the operators exactly for ROM dimension $n = 12$. In [101], 10^5 FOM time steps are used to infer ROM operators for ROM dimension $n = 14$ from a regularized operator inference minimization problem that significantly differ from the intrusive counterparts.

Note that $\mathbf{A}_2 = \mathbf{0}$ in our discretization, so we could omit this term in the operator inference, so we would even need $n_2 = 105$ less FOM time steps. We refrain from this simplification to keep the comparison with [77] and [101] fair.

3.4.2 Ice sheet model

We consider the shallow ice equations in a setup based on [6],

$$\begin{aligned} \frac{\partial}{\partial t} x(\xi, t) = & c_1 \frac{\partial}{\partial \xi} \left(x^2(\xi, t) \frac{\partial}{\partial \xi} x(\xi, t) \right) \\ & + c_2 \frac{\partial}{\partial \xi} \left(x^5(\xi, t) \left| \frac{\partial}{\partial \xi} x(\xi, t) \right|^2 \frac{\partial}{\partial \xi} x(\xi, t) \right) \end{aligned} \quad (3.31)$$

with the spatial coordinate $\xi \in [0, 1000]$, time $t \in [0, 2]$ and the constants $c_1 = 8.9 \cdot 10^{-13}$ and $c_2 = 2.8 \cdot 10^7$. We impose the initial condition

$$x(\xi, 0) = 10^{-2} + 630 \left(\frac{\xi}{2000} + 0.25 \right)^4 \left(\frac{\xi}{2000} - 0.75 \right)^4, \quad (3.32)$$

and homogeneous Neumann boundary conditions. We discretize with finite differences in space with mesh width $\Delta \xi = \frac{1000}{29}$, and with implicit Euler in time with time step size $\Delta t = 10^{-3}$, to obtain the time-discrete dynamical system of dimension $N = 512$,

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \Delta t \left(\mathbf{A}_3 \mathbf{x}_k^3 + \mathbf{A}_8 \mathbf{x}_k^8 \right), \quad k = 0, \dots, K-1. \quad (3.33)$$

This system is polynomial of degree $l = 8$ with $\mathbf{A}_3 \in \mathbb{R}^{N \times N_3}$ with $N_3 = \frac{N(N+1)(N+2)}{6}$ and $\mathbf{A}_8 \in \mathbb{R}^{N \times N_8}$ with $N_8 = \binom{N+7}{8}$ and $K = 2000$. The resulting 2001 snapshots from $t = 0$ to $t = 2$ are used to compute the POD bases $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ with $n = 1, \dots, 7$.

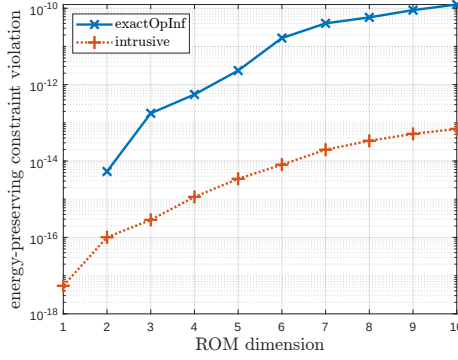


Figure 3.3: Violation of the energy-preserving constraint of the convection operator of Burgers' equation.

We apply Algorithm 3.1 with $\Delta t = 1.4773 \cdot 10^{13}$ according to estimate (3.25). This extreme value can be explained by the extreme operator norm $\|\tilde{\mathbf{O}}\|_2 = 4.271 \cdot 10^{-14}$ for ROM dimension $n = 1$. The relative operator errors for ROM dimensions $n = 1, \dots, 7$ are depicted in Fig. 3.1. Like in the previous test case, these operator errors are essentially on the order of machine precision. The fact that these errors are a few orders of magnitude larger than the errors in the previous test case is presumably related to the larger condition numbers as depicted in Fig. 3.2.

For ROM dimension $n = 7$, the proposed method performs $n_f = 3087$ FOM time steps. This number is larger than the 2001 FOM time steps with re-projection reported in [6]. However, the authors also report that this budget of snapshots is depleted before the inference of the ROM operators of dimension $n = 6$ is completed. Hence, their ROM operators of dimensions $n = 6$ and $n = 7$ are partially computed with regularized operator inference and snapshot data polluted with non-Markovian terms. Consequently, their inferred operators of dimension $n = 6$ and $n = 7$ are not reconstructed exactly. Their operators of dimension $n \leq 5$ might be reconstructed exactly, but the used 2001 snapshots are more than the $n_f = 365$ snapshots used by our method for dimension $n = 5$.

Note that the cubic term could be neglected in practice due to the extremely small value of c_1 . This simplification would reduce the number of required FOM time steps further.

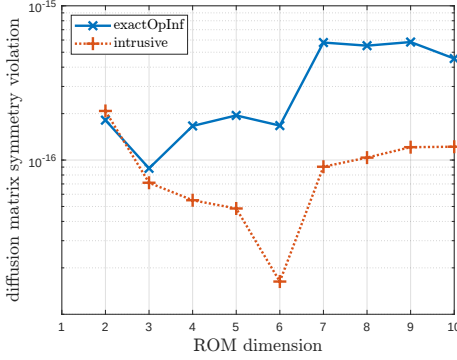


Figure 3.4: Symmetry violation of the diffusion matrix of Burgers' equation.

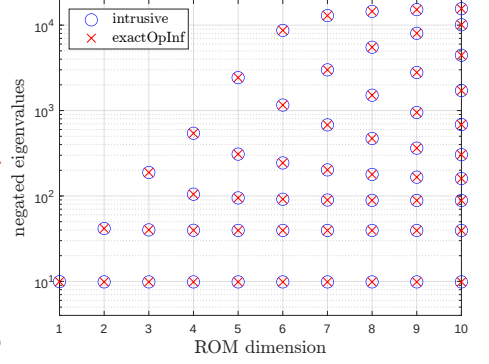


Figure 3.5: Eigenvalues of diffusion matrix of Burgers' equation multiplied with -1 .

3.4.3 Burgers' equation

We consider the viscous Burgers' equation in a setup based on [62, 77],

$$\frac{\partial}{\partial t} x(\xi, t) + x(\xi, t) \frac{\partial}{\partial \xi} x(\xi, t) - \frac{\partial^2}{\partial \xi^2} x(\xi, t) = 0, \quad (3.34)$$

with spatial coordinate $\xi \in \Omega = (-1, 1)$, time $t \in [0, 1]$. We impose periodic boundary conditions and the initial condition $x(\xi, 0) = -\sin(\pi\xi)$ for all $\xi \in \Omega$.

We discretize with finite differences on an equidistant grid in space with mesh width $\Delta\xi = 2^{-6}$, and with explicit Euler in time with step size $\Delta t = 10^{-4}$, to obtain the time-discrete dynamical system of dimension $N = 128$,

$$x_{k+1} = x_k + \Delta t \left(\mathbf{A}_1 x_k + \mathbf{A}_2 x_k^2 \right), \quad k = 0, \dots, K-1. \quad (3.35)$$

This system is polynomial of degree $l = 2$ with $\mathbf{A}_1 \in \mathbb{R}^{N \times N}$, $\mathbf{A}_2 \in \mathbb{R}^{N \times N_2}$ with $N_2 = \frac{N(N+1)}{2}$ and $K = 10^4$. Concretely, we use a spatial discretization as described in [41] such that the diffusion matrix $\mathbf{A}_1$ is symmetric and negative semi-definite, and the convection operator $\mathbf{A}_2$ satisfies the skew-symmetry property $x^T \mathbf{A}_2 x^2 = 0$ for all $x \in \mathbb{R}^N$. The resulting $10^4 + 1$ snapshots are used to compute the POD bases $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ with $n = 1, \dots, 10$.

We perform Algorithm 3.1 with $\Delta t = 1.0134 \cdot 10^{-1}$ according to estimate (3.25) and compute the relative operator errors for ROM dimensions $n = 1, \dots, 10$. Fig. 3.1 shows operator errors on the order of machine precision and Fig. 3.2 very small condition numbers. In addition, we observe that both, the intrusive and the inferred operators preserve the skew-symmetry of the FOM convection operator

and the symmetry and negative semi-definiteness of the FOM diffusion operator. Fig. 3.3 shows the energy-preserving constraint violation $\sum_{1 \leq i,j,k \leq n} |\hat{h}_{ijk} + \hat{h}_{jik} + \hat{h}_{kji}|$ as defined in [62] for the inferred operator with $\hat{h}_{ijk} = \mathbf{e}_i \hat{\mathbf{A}}_2 \mathbf{I}_N^{(2)}(\mathbf{e}_j \otimes \mathbf{e}_k)$ and for the intrusive operator with $\hat{h}_{ijk} = \mathbf{e}_i \tilde{\mathbf{A}}_2 \mathbf{I}_N^{(2)}(\mathbf{e}_j \otimes \mathbf{e}_k)$. As can be seen, this violation is essentially on the order of machine precision for both, inferred and intrusive operators. Fig. 3.4 shows the symmetry violation of the inferred and the intrusive diffusion matrix, $\frac{\|\hat{\mathbf{A}}_1 - \hat{\mathbf{A}}_1^T\|}{\|\hat{\mathbf{A}}_1\|}$ and $\frac{\|\tilde{\mathbf{A}}_1 - \tilde{\mathbf{A}}_1^T\|}{\|\tilde{\mathbf{A}}_1\|}$, respectively. Also this violation is on the order of machine precision. Consequently, the diffusion matrices are symmetric, so their eigenvalues are real. These eigenvalues — multiplied with -1 — are depicted in Fig. 3.5. As can be seen the eigenvalues of the inferred diffusion matrices coincide with the eigenvalues of the corresponding intrusive matrices. Hence, both ROMs in particular preserve the negative semi-definiteness of the FOM diffusion operator.

3.5 CONCLUSION

This chapter introduces a method to generate rank-ensuring snapshot data for operator inference that guarantees the exact reconstruction of the corresponding intrusive reduced order model. This method performs the minimal number of FOM time steps and yields a square data matrix, so the operator inference least squares problem can be simplified to a linear system that has better numerical stability. Numerical experiments confirm that the novel snapshot data generation method leads to exact reconstruction of the intrusive ROM operators. Consequently, also properties such as skew-symmetry, symmetry and negative semi-definiteness of the intrusive operators are preserved by the non-intrusive operators without the need to enforce this structure preservation explicitly.

To improve the condition number of the matrix corresponding to the operator inference linear system, future work could investigate the exploitation of nestedness to solve subproblems sequentially [6] or different choices of rank-ensuring ROM states, for example based on Chebychev points [107]. Future work should also investigate how to enable that the proposed rank-ensuring states can be specified as initial conditions in solvers that allow only a certain range of state values or require specific boundary conditions.

In this chapter, we have not considered the parameter dependency of ROM operators. For that purpose, the method proposed in this chapter is combined with [70] to generate rank-sufficient operator inference snapshot data for parametric problems in Chapter 4.

EXACT OPERATOR INFERENCE DECOUPLES PARAMETRIC PROBLEMS

This chapter presents an advance in the field of parametric model reduction. The application of operator inference to non-intrusive parametric problems whose operators exhibit affine parameter dependency results in a minimization problem with three issues. First, the solution to the minimization problem is generally not unique, so we might obtain a different solution than the one we are interested in. Second, time discretization inconsistencies and non-Markovian information can cause errors in the inferred operators. Third, the large dimensions of the minimization problem can cause prohibitively high computational costs.

In this chapter, we resolve these issues by extending previous work on exact operator inference to the parametric case. Concretely, we propose a snapshot data generation method that guarantees the uniqueness of the solution and the absence of time discretization inconsistencies and non-Markovian information, and thereby the exact reconstruction of the corresponding intrusive operators of projection-based reduced order models. Furthermore, we show that the monolithic problem that comprises the snapshot data of all sampled parameters can be decoupled into separate problems for only one parameter sample each. This decoupling results in smaller subproblems that are embarrassingly parallel, numerically more stable and can be solved more efficiently.

In numerical experiments, we demonstrate this decoupling for the heat equation with two different temperature-, parameter- and space-dependent thermal conductivities.

4.1 INTRODUCTION

The aim of model reduction is to reduce the computational cost of numerical simulations in complex engineering tasks such as optimization, data assimilation, uncertainty quantification and the development of digital twins. All these tasks involve multiple simulations that depend on variable parameters.

In this chapter we consider problems described by a dynamical system with parameter-dependent right-hand side. Such dynamical systems are, for example, derived from spatial discretizations of PDEs with parameter-dependent geometries or physical properties. Other types of parameter-dependency considered in other works include

*This chapter is based on the following article:
Henrik Rosenberger,
Benjamin Sanderse, and
Giovanni Stabile. "Exact
operator inference
decouples parametric
problems." In: Computer
Methods in Applied
Mechanics and
Engineering 461 (2026),
p. 119199. DOI: [10.1016/
j.cma.2026.119199](https://doi.org/10.1016/j.cma.2026.119199).*

**CRedit author
statement**
Henrik Rosenberger:
Conceptualization,
Methodology, Software,
Writing – original draft.
Benjamin Sanderse:
Funding acquisition,
Writing – review &
editing.
Giovanni Stabile:
Writing – review &
editing.

parametric initial conditions [62, 67] and parametric inputs such as boundary conditions [15].

In intrusive projection-based model reduction, parameter dependencies in the operators of the high-fidelity model are directly inherited by the reduced-order model (ROM) [12]. However, intrusive model reduction is often not feasible in practice because it requires access to numerical operators that is not possible in industry software. To address this practical problem, nonintrusive model reduction constructs ROMs without intrusive software access but only based on output data of the high-fidelity simulations. In particular, operator inference aims at reproducing the same ROMs as intrusive projection-based model reduction but nonintrusively [63, 78]. Based on snapshot data of high-fidelity simulations, ROM operators of given polynomial degree are computed as solution to a minimization problem. However, in context of parameter-dependent problems [34, 70, 113, 118, 119], operator inference has three issues. First, the solution to the minimization problem is not unique if the provided snapshot data is not informative enough. In that case the computed solution can be very different to the intrusive ROM operators that we are interested in. Second, time discretization inconsistencies and non-Markovian information in the snapshot data can introduce errors in the inferred operators. This issue has been resolved in [77] via a method known as re-projection that guarantees exact reconstruction if the first issue does not take effect. Third, solving the minimization problem can cause prohibitively high computational costs because of the large dimensions of this problem. These dimensions depend on the complexity of the parameter dependency, the polynomial degree of the underlying problem and the considered ROM dimension. If these quantities are sufficiently large, the computational costs of the minimization problem can even dominate the overall offline costs of the ROM.

In this chapter we resolve all three issues by extending previous work on non-parametric operator inference [92] to the parametric case. Concretely, we propose a snapshot data generation method that guarantees the uniqueness of the solution to the minimization problem and avoids time discretization inconsistencies and non-Markovian information in the snapshot data. As a consequence, the computed solution is an exact reconstruction of the intrusive ROM operators up to machine precision. Moreover, we show that for this snapshot data the minimization problem simplifies to a linear system that decouples into smaller linear systems. As a result, the computational costs are significantly smaller. In addition, the decoupled linear systems have a smaller condition number and can be solved in parallel.

This chapter is organized as follows. In Section 4.2, we introduce parametric dynamical systems, intrusive and non-intrusive projection-based model reduction and non-parametric exact operator inference. In

Section 4.3 we extend exact operator inference to the parametric case. In Section 4.4 we describe the decoupling resulting from this extension. In Section 4.5 we present an alternative derivation for the same decoupling method. In Section 4.6 we present numerical experiments for the heat equation with different thermal conductivities. Finally, in Section 4.7 we draw conclusions.

4.2 PRELIMINARIES

4.2.1 Parametric dynamical system

We consider the dynamical system of dimension N ,

$$\dot{x}(t; \mu) = f(x(t; \mu); \mu), \quad x(0; \mu) = x_0(\mu), \quad (4.1)$$

parametrized by $\mu \in \mathcal{D} \subset \mathbb{R}^d$. We assume that the function $f : \mathbb{R}^N \times \mathcal{D} \rightarrow \mathbb{R}^N$ has polynomial structure, and for ease of presentation, we assume a quadratic term here,

$$f(x; \mu) = \mathbf{H}(\mu)x^2, \quad (4.2)$$

where $x^2 \in \mathbb{R}^{\binom{N+1}{2}}$ comprises all unique products of two entries of x as described in [77, 92] and $\mathbf{H} : \mathcal{D} \rightarrow \mathbb{R}^{N \times \binom{N+1}{2}}$. Functions f that are monomials in x of any other order can be handled analogously. Functions f that are sums of multiple monomial terms are discussed in Appendix C.1.

4.2.2 Parametric Galerkin ROM

To obtain a Galerkin ROM, the FOM is projected onto a basis $\mathbf{V}_n \in \mathbb{R}^{N \times n}$, $n \ll N$, and the FOM state $x(t; \mu)$ is replaced by a lower dimensional approximation $\mathbf{V}_n \tilde{x}(t; \mu)$,

$$\dot{\tilde{x}}(t; \mu) = \tilde{\mathbf{H}}(\mu) \tilde{x}^2(t; \mu) \quad (4.3)$$

with the ROM state $\tilde{x}(t; \mu) \in \mathbb{R}^n$ and the projected matrix defined by

$$\tilde{\mathbf{H}}(\mu) = \mathbf{V}_n^T \mathbf{H}(\mu) \mathbf{I}_N^{(i)} (\mathbf{V}_n \otimes \mathbf{V}_n) \mathbf{J}_n^{(i)}, \quad (4.4)$$

with matrices $\mathbf{I}_N^{(i)} \in \{0, 1\}^{\binom{N+1}{2} \times N^2}$ and $\mathbf{J}_n^{(i)} \in \{0, 1\}^{n^2 \times \binom{n+1}{2}}$ as described in [77].

There are several methods to define the basis $\mathbf{V}_n$ [12]. In this chapter, we consider a global basis obtained from POD of snapshots

$$[\mathbf{X}(\bar{\mu}_1) \quad \dots \quad \mathbf{X}(\bar{\mu}_{s_{\text{POD}}})]$$

with

$$\mathbf{X}(\bar{\boldsymbol{\mu}}_i) := [\mathbf{x}(\bar{t}_1; \bar{\boldsymbol{\mu}}_i) \ \dots \ \mathbf{x}(\bar{t}_{K_{\text{POD}}}; \bar{\boldsymbol{\mu}}_i)] \in \mathbb{R}^{N \times K_{\text{POD}}}$$

at sampled parameters $\bar{\boldsymbol{\mu}}_1, \dots, \bar{\boldsymbol{\mu}}_{s_{\text{POD}}} \in \mathcal{D}$ with $K_{\text{POD}}, s_{\text{POD}} \in \mathbb{N}$. How to choose such parameter samples is discussed in [12].

Computing ROM operators for new parameter values $\boldsymbol{\mu}$ via (4.4) is not efficient because the computational costs scale with the FOM dimension N . Instead, we assume that the FOM operator can be affinely decomposed,

$$\mathbf{H}(\boldsymbol{\mu}) = \sum_{p=1}^{q_H} \theta_H^{(p)}(\boldsymbol{\mu}) \mathbf{H}^{(p)}, \quad (4.5)$$

with scalar functions $\theta_H^{(p)} : \mathcal{D} \rightarrow \mathbb{R}$ and matrices $\mathbf{H}^{(p)} \in \mathbb{R}^{N \times \binom{N+1}{2}}$, $p = 1, \dots, q_H$. Then, we can precompute

$$\tilde{\mathbf{H}}^{(p)} = \mathbf{V}_n^T \mathbf{H}^{(p)} \mathbf{I}_N^{(i)} (\mathbf{V}_n \otimes \mathbf{V}_n) \mathbf{J}_n^{(i)}, \quad p = 1, \dots, q_H, \quad (4.6)$$

during the offline phase, and compute

$$\tilde{\mathbf{H}}(\boldsymbol{\mu}) = \sum_{p=1}^{q_H} \theta_H^{(p)}(\boldsymbol{\mu}) \tilde{\mathbf{H}}^{(p)} \quad (4.7)$$

efficiently during the online phase with computational costs only depending on the ROM dimension n . However, the computations (4.6) are intrusive since they require access to the FOM operator components $\mathbf{H}^{(r)}$. As a non-intrusive alternative, our goal is to infer the ROM operator components $\tilde{\mathbf{H}}^{(r)}$ from data in the next section.

Note that not all parametric FOM operators possess an affine structure of the form (4.5) with known coefficient functions $\theta_H^{(p)} : \mathcal{D} \rightarrow \mathbb{R}$. However, operators without this structure can still be approximated by such affine expansions. For example, Taylor approximations [20], interpolation [30] and hyperreduction techniques such as DEIM [12] and GNAT [24] have been used in the context of intrusive model reduction. These hyperreduction techniques are based on the assumption that single entries of operators can be computed more efficiently than the complete operators. Non-intrusive model reduction, however, does not directly access operators at all, so hyperreduction techniques are not applicable. In the non-intrusive context of operator inference, affine expansions have been considered that describe the parameter dependency exactly [70, 113] or are derived from Taylor approximations [34] and piecewise-linear interpolation [60]. All these approximations introduce errors and can suffer from the curse of dimensionality if $\boldsymbol{\mu}$ is high-dimensional, so achieving satisfactory accuracy can be challenging [12].

4.2.3 Parametric operator inference

As in the non-parametric case, operator inference with affine parameter dependency uses two sets of snapshot data. The first one to generate the ROM basis, the second one to infer the operators. These two sets can comprise different snapshots; in particular, parameter samples the $\mu_1, \dots, \mu_s \in \mathcal{D}$ corresponding to the second set can be different from the parameter samples $\bar{\mu}_1, \dots, \bar{\mu}_{s_{\text{POD}}}$ corresponding to the first set, as described in the previous subsection. Let

$$\mathbf{X}(\mu_i) := [\mathbf{x}(t_1; \mu_i) \quad \dots \quad \mathbf{x}(t_K; \mu_i)] \in \mathbb{R}^{N \times K}, \quad i = 1, \dots, s \quad (4.8)$$

be this second set of snapshots with the corresponding time derivative snapshots

$$\dot{\mathbf{X}}(\mu_i) := \left[\frac{d}{dt} \mathbf{x}(t_1; \mu_i) \quad \dots \quad \frac{d}{dt} \mathbf{x}(t_K; \mu_i) \right] \in \mathbb{R}^{N \times K}, \quad i = 1, \dots, s, \quad (4.9)$$

with $K, s \in \mathbb{N}$ not necessarily equal to K_{POD} and s_{POD} , respectively.

As described in [70, 119], the operator inference problem to infer the operator matrix

$$\hat{\mathbf{O}} := [\hat{\mathbf{H}}^{(1)} \quad \dots \quad \hat{\mathbf{H}}^{(q_H)}] \in \mathbb{R}^{n \times q(n)}, \quad q(n) = q_H \binom{n+1}{2} \quad (4.10)$$

from the projected snapshot data

$$\hat{\mathbf{X}}(\mu_i) = \mathbf{V}^T \mathbf{X}(\mu_i), \quad i = 1, \dots, s, \quad (4.11)$$

$$\hat{\dot{\mathbf{X}}}(\mu_i) = \mathbf{V}^T \dot{\mathbf{X}}(\mu_i), \quad i = 1, \dots, s, \quad (4.12)$$

is to minimize

$$\mathcal{L}(\hat{\mathbf{O}}) := \sum_{i=1}^s \left\| \hat{\mathbf{X}}_2(\mu_i)^T \left[\sum_{p=1}^{q_H} \theta_H^{(p)}(\mu_i) \hat{\mathbf{H}}^{(p)} \right]^T - \hat{\dot{\mathbf{X}}}(\mu_i)^T \right\|_F^2 \quad (4.13)$$

$$= \left\| \mathbf{D} \hat{\mathbf{O}}^T - \mathbf{R}^T \right\|_F^2. \quad (4.14)$$

with the matrices $\mathbf{R} = [\hat{\mathbf{X}}(\mu_1) \quad \dots \quad \hat{\mathbf{X}}(\mu_s)] \in \mathbb{R}^{n \times sK}$ and

$$\mathbf{D} := \begin{bmatrix} \boldsymbol{\theta}_H(\mu_1) \otimes \hat{\mathbf{X}}_2(\mu_1)^T \\ \vdots \\ \boldsymbol{\theta}_H(\mu_s) \otimes \hat{\mathbf{X}}_2(\mu_s)^T \end{bmatrix} \in \mathbb{R}^{sK \times q_H \binom{n+1}{2}}, \quad (4.15)$$

with

$$\theta_H(\mu_i) = \begin{bmatrix} \theta_H^{(1)}(\mu_i) & \dots & \theta_H^{(q_H)}(\mu_i) \end{bmatrix} \in \mathbb{R}^{1 \times q_H}, \quad i = 1, \dots, s, \quad (4.16)$$

and

$$\hat{\mathbf{X}}_2(\mu_i) := \begin{bmatrix} \hat{x}^2(t_1; \mu_i) & \dots & \hat{x}^2(t_K; \mu_i) \end{bmatrix} \in \mathbb{R}^{\binom{n+1}{2} \times K}. \quad (4.17)$$

There are three issues with the operator inference problem of minimizing (4.14): First, the uniqueness of the solution to this minimization problem is generally not guaranteed. Consequently, we might obtain a different solution than the one we are interested in. Second, time discretization inconsistencies and non-Markovian information in the snapshot data can introduce errors in the inferred operators. Both issues have been addressed by us in [92] in the context of non-parametric operator inference. In the next subsection, we will summarize that work and in Section 4.3 we will show how it can be generalized to the parametric case. Third, this minimization problem has very large dimensions. These large dimensions cause high computational costs and can affect numerical stability [119]. In detail, the matrix $\mathbf{D}$ has column dimension $q_H \binom{n+1}{2}$ or more generally a column dimension of $\mathcal{O}(q_l n^l)$ with l the largest polynomial degree in the right-hand side $\mathbf{f}$ in (4.1) and q_l the number of terms in the affine decomposition analogous to (4.5) of the operator with this highest polynomial degree l . The computational cost of solving the least squares problem scales cubically in this column dimension of $\mathbf{D}$, so $\mathcal{O}((q_l n^l)^3)$. These costs can contribute substantially to the offline preparation of the ROM, even though they do not depend on the FOM dimension N . Moreover, the computational costs can be even larger when the linear least squares problem (4.14) is extended by constraints [16, 101, 103] or replaced by a nonlinear optimization problem [45] in order to preserve structure. The large dimensions are a consequence of comprising all affine operators $\hat{\mathbf{H}}^{(i)}$ and the snapshot data for all parameter samples μ_i in one single optimization problem. In contrast, in intrusive model reduction, the affine operators $\hat{\mathbf{H}}^{(i)}$ can be computed separately as described in (4.6). To achieve a similar separation with non-intrusive operator inference, we present in Section 4.4 a strategy to decouple the monolithic optimization problem (4.14) into smaller subproblems that only pertain one affine operator $\hat{\mathbf{H}}^{(i)}$ each. These smaller problems can be solved embarrassingly parallel and have a smaller condition number and are hence numerically more stable.

*Solving the monolithic
parametric operator
inference problem can
dominate the ROM
offline costs.*

4.2.4 Exact operator inference

In [92], we have proposed a method to generate snapshot data for operator inference that guarantees exact reconstruction in the non-parametric case. This method has two key steps: First we identify

ROM states that ensure the full rank of the operator inference least squares matrix. This identification is done based on the polynomial terms contained in the right-hand side f in (4.1). In the purely quadratic case (4.2) considered here, such a set of ROM states is given by the set of all sums of two unit vectors in $\mathbb{R}^n$,

$$\mathcal{X}^2 := \{e_i + e_j \mid e_i, e_j \in \mathbb{R}^n \text{ unit vectors}\}. \quad (4.18)$$

Let $\hat{x}_j \in \mathcal{X}^2, j = 1, \dots, \binom{n+1}{2}$ be the $\binom{n+1}{2}$ distinct elements of $\mathcal{X}^2$, then the matrix

$$\hat{\mathbf{X}}_2 := \begin{bmatrix} \hat{x}_1^2 & \dots & \hat{x}_{\binom{n+1}{2}}^2 \end{bmatrix} \in \mathbb{R}^{\binom{n+1}{2} \times \binom{n+1}{2}} \quad (4.19)$$

has full rank.

To generate the matrix $\hat{\mathbf{X}}$ of corresponding ROM state time derivative snapshots, the second key step is to perform $\binom{n+1}{2}$ FOM simulations each for only a single time step. The initial conditions of these simulations are the FOM states corresponding to the identified ROM states $\hat{x}_j$.

The snapshot data generated by this method is equivalent to snapshot data that one would obtain from simulating the corresponding intrusive ROM. Consequently, the intrusive ROM operator is a solution to the operator inference least squares problems. Since the matrix $\hat{\mathbf{X}}_2$ has full rank, this solution is unique, so operator inference exactly reconstructs the intrusive ROM operators.

The complete snapshot data generation method is summarized in Algorithm 4.1. In exact arithmetics, the time step size Δt is irrelevant for the resulting matrices $\hat{\mathbf{X}}_2$ and $\hat{\mathbf{X}}$. In practice, however, Δt should be chosen to enable accurate floating-point number representations as explained in more detail in [92].

4.3 EXACT PARAMETRIC OPERATOR INFERENCE

In this section we address the issue of non-unique solutions to the parametric operator inference least squares problem defined by (4.14). This problem has a unique solution if the matrix $\mathbf{D}$ has full column rank. As described in Theorem 2.3 in [70], $\mathbf{D}$ has full column rank if the matrix $\Theta_H \in \mathbb{R}^{s \times q_H}$ defined by

$$[\Theta_H]_{ij} = \theta_H^{(j)}(\mu_i), \quad (4.20)$$

and all $\hat{\mathbf{X}}_2(\mu_i), i = 1, \dots, s$ have full column rank.

The first novelty of this chapter is a snapshot data generation method that guarantees full rank of the matrices $\hat{\mathbf{X}}_2(\mu_i), i = 1, \dots, s$. For this

Algorithm 4.1 Exact operator inference snapshot data generation [92]

```

1: procedure EXACTOPINF Snapshots(
    ROM basis  $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ ,
    time step size  $\Delta t$ 
)
2:   Initialize  $\hat{\mathbf{X}}_2 \leftarrow \emptyset$ 
3:   Initialize  $\hat{\mathbf{X}} \leftarrow \emptyset$ 
4:   for  $\hat{\mathbf{x}} \in \mathcal{X}^2$  do
5:      $\mathbf{x}_0 \leftarrow \mathbf{V}_n \hat{\mathbf{x}}$ 
6:      $\mathbf{x}_1 \leftarrow \text{FOM}(\text{initial condition } \mathbf{x}_0, \text{time step size } \Delta t,$ 
        end time  $T = \Delta t,$ 
        explicit Euler time discretization )
7:      $\hat{\mathbf{X}}_2 \leftarrow \hat{\mathbf{x}}^2$ 
8:      $\hat{\mathbf{X}} \leftarrow [\hat{\mathbf{X}}, \frac{\mathbf{V}_n^T \mathbf{x}_1 - \hat{\mathbf{x}}}{\Delta t}]$ 
9:   end for
10:  return  $\hat{\mathbf{X}}_2, \hat{\mathbf{X}}$ 
11: end procedure

```

The extension of exact operator inference to parametric problems is straightforward.

purpose we use the exact non-parametric operator inference snapshot data generation method described in Section 4.2.4 and extend it to the parametric case. In detail, we run Algorithm 4.1 for each parameter sample μ_i to obtain the outputs $\hat{\mathbf{X}}_2(\mu_i)$ and $\hat{\mathbf{X}}(\mu_i)$ as summarized in Algorithm 4.2. Note that the matrices $\hat{\mathbf{X}}_2(\mu_i)$ are actually not affected by the parameter μ_i but equal to the matrix $\hat{\mathbf{X}}_2$ resulting from the non-parametric Algorithm 4.1. In particular these matrices all have full column rank.

On the other hand, ensuring full column rank of Θ_H is highly problem-specific. In view of the definitions of the functions $\theta_H^{(j)}$, parameter samples μ_i must be chosen to achieve this full rank. In general, it is unclear how such parameter samples can be found; this is beyond the scope of this thesis. However, for polynomial functions $\theta_H^{(j)}$ that arise for example from Taylor approximations as described in [34], full-rank-ensuring parameter samples can be identified analogously to the full-rank-ensuring ROM states in [92].

If Θ_H indeed has full column rank and the parameter dependency of the operator $\hat{\mathbf{H}}$ can be expressed exactly by an affine expansion such as (4.5), then the snapshot data generated proposed in [92] ensures that the parametric operator inference exactly reconstructs the intrusive ROM operators. Furthermore, for any full column rank matrix $\Theta_H \in \mathbb{R}^{s \times q_H}$ with $s > q_H$, there exists a subset of q_H rows that also form a matrix with full column rank q_H . These rows correspond to q_H of the s parameter

Algorithm 4.2 Exact parametric operator inference snapshot data generation

```

1: procedure EXACTPARAMETRICOPINFSNAPSHOTS(
    ROM basis  $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ ,
    time step size  $\Delta t$ 
    parameter samples  $\mu_1, \dots, \mu_s$ 
  )
2:   for  $\mu_i = \mu_1, \dots, \mu_s$  do
3:     Initialize  $\hat{\mathbf{X}}_2(\mu_i) \leftarrow \emptyset$ 
4:     Initialize  $\hat{\dot{\mathbf{X}}}(\mu_i) \leftarrow \emptyset$ 
5:     for  $\hat{x} \in \mathcal{X}^2$  do
6:        $\mathbf{x}_0 \leftarrow \mathbf{V}_n \hat{x}$ 
7:        $\mathbf{x}_1 \leftarrow \text{FOM}(\text{initial condition } \mathbf{x}_0, \text{time step size } \Delta t,$ 
          end time  $T = \Delta t,$ 
          explicit Euler time discretization
          parameter  $\mu_i$  )
8:        $\hat{\mathbf{X}}_2(\mu_i) \leftarrow \hat{x}^2$ 
9:        $\hat{\dot{\mathbf{X}}}(\mu_i) \leftarrow [\hat{\dot{\mathbf{X}}}(\mu_i), \frac{\mathbf{V}_n^T \mathbf{x}_1 - \hat{x}}{\Delta t}]$ 
10:    end for
11:  end for
12:  return  $(\hat{\mathbf{X}}_2(\mu_1), \hat{\dot{\mathbf{X}}}(\mu_1)), \dots, (\hat{\mathbf{X}}_2(\mu_s), \hat{\dot{\mathbf{X}}}(\mu_s))$ 
13: end procedure

```

samples μ_i , $i = 1, \dots, s$. In the following, we can therefore assume that $s = q_H$ parameter samples are used.

Additionally, as described in [92], Algorithm 4.1 does not introduce time discretization inconsistencies or non-Markovian information in the snapshot data. This property is preserved by Algorithm 4.2, so the generated snapshots guarantee that the solution to the parametric operator inference problem (4.14) reconstructs the intrusive ROM operators exactly. The following section describes how the computational costs of solving this problem can be reduced significantly via decoupling.

4.4 DECOUPLING OF EXACT PARAMETRIC OPERATOR INFERENCE

The parametric operator inference minimization problem (4.14) is high-dimensional because it involves data from $s = q_H$ parameter samples to learn q_H quadratic operators. We want to decouple this problem into q_H subproblems that each only involve data from a single parameter sample to learn a single quadratic operator. For this purpose, we first note that if Θ_H is the s -dimensional identity matrix $\mathbf{I}_s$, then the objective function (4.13) simplifies to

$$\mathcal{L}(\hat{\mathbf{O}}) = \sum_{i=1}^s \left\| \hat{\mathbf{X}}_2(\mu_i)^T \left[\theta_H^{(i)}(\mu_i) \hat{\mathbf{H}}^{(i)} \right]^T - \hat{\mathbf{X}}(\mu_i)^T \right\|_F^2, \quad (4.21)$$

so for all $i = 1, \dots, s$, the operator $\hat{\mathbf{H}}^{(i)}$ only depends on data from the parameter sample μ_i . Hence, one can solve subproblems for the basis operators $\hat{\mathbf{H}}^{(i)}$ separately,

$$\hat{\mathbf{H}}^{(i)} = \min_{\hat{\mathbf{H}}^{(i)}} \left\| \hat{\mathbf{X}}_2(\mu_i)^T \left[\theta_H^{(i)}(\mu_i) \hat{\mathbf{H}}^{(i)} \right]^T - \hat{\mathbf{X}}(\mu_i)^T \right\|_F^2, \quad i = 1, \dots, q_H. \quad (4.22)$$

In general, however, the condition $\Theta_H = \mathbf{I}_s$ does not hold. There are special cases such as in [60] where the condition $\Theta_H = \mathbf{I}_s$ is implicitly satisfied because the parameter dependency of a linear operator is approximated by piecewise-linear interpolation of the operator evaluated at parameter samples. In other cases, parameter samples μ_i might exist such that $\Theta_H = \mathbf{I}_s$ as well. However, for most affine expansions (4.5) such parameter samples do not exist, or are not physical, so corresponding snapshot data cannot be generated. In the following, we therefore propose a decoupling method for general square matrices Θ_H with full rank.

This decoupling method is based on two key insights. The first key insight is that the least squares problem defined by (4.14) can be simplified to a linear system analogously to the non-parametric case in

[92]. For this purpose, we first note that this least squares problem is equivalent to the normal equations

$$\mathbf{D}^T \mathbf{D} \hat{\mathbf{D}}^T = \mathbf{D}^T \mathbf{R}^T. \quad (4.23)$$

As described in Section 4.3, the matrix $\mathbf{D}$ has full rank if the matrices $\hat{\mathbf{X}}_2(\mu_i)$ are obtained from Algorithm 4.2 with suitable parameter samples μ_i , $i = 1, \dots, s$. If additionally $s = q_H$, then $\mathbf{D}$ is also square and the normal equations simplify to

$$\mathbf{D} \hat{\mathbf{D}}^T = \mathbf{R}^T. \quad (4.24)$$

The second key insight of our method is that, as described in Section 4.3, the matrices $\hat{\mathbf{X}}_2(\mu_i)$ resulting from Algorithm 4.2 are actually independent of the parameter μ_i and equal to the matrix $\hat{\mathbf{X}}_2$ resulting from the non-parametric Algorithm 4.1. Using $\hat{\mathbf{X}}_2(\mu_i) = \hat{\mathbf{X}}_2$, the matrix $\mathbf{D}$ defined in (4.15) simplifies to

$$\mathbf{D} = [\Theta_H \otimes \hat{\mathbf{X}}_2^T], \quad (4.25)$$

with Θ_H as defined in (4.20). This simplified structure allows us to use preconditioning to transform the linear system (4.24) into a linear system for the matrix $\check{\mathbf{D}} = [\mathbf{I}_{q_H} \otimes \hat{\mathbf{X}}_2^T]$. For the monomial right-hand side f considered here, both left- and right-preconditioning are possible. For non-monomial right-hand sides f , only right-preconditioning is possible as described in Appendix C.1. For this reason, we also present the right-preconditioning here. We substitute

$$\hat{\mathbf{O}}^T = [\Theta_H^{-1} \otimes \mathbf{I}_{\binom{n+1}{2}}] \check{\mathbf{O}}^T \quad (4.26)$$

to find

$$\begin{aligned} \mathbf{D} \hat{\mathbf{O}}^T &= [\Theta_H \otimes \hat{\mathbf{X}}_2^T] [\Theta_H^{-1} \otimes \mathbf{I}_{\binom{n+1}{2}}] \check{\mathbf{O}}^T = [\mathbf{I}_{q_H} \otimes \hat{\mathbf{X}}_2^T] \check{\mathbf{O}}^T \\ &=: \check{\mathbf{D}} \check{\mathbf{O}}^T. \end{aligned} \quad (4.27)$$

Then, the linear system $\check{\mathbf{D}} \check{\mathbf{O}}^T = \mathbf{R}^T$ is decoupled and can be written as

$$\hat{\mathbf{X}}_2^T \left[\check{\mathbf{H}}^{(i)} \right]^T = \hat{\mathbf{X}}(\mu_i)^T, \quad i = 1, \dots, q_H. \quad (4.28)$$

After solving for the operators $\check{\mathbf{H}}^{(i)}$, the original basis operators are then obtained via (4.26),

$$\hat{\mathbf{H}}^{(i)} = \sum_{j=1}^{q_H} [\Theta_H^{-1}]_{ij} \check{\mathbf{H}}^{(j)}, \quad i = 1, \dots, q_H. \quad (4.29)$$

*Exact operator inference
state snapshots are
parameter-independent.*

Note that this transformation involves the explicit computation of the inverse Θ_H^{-1} . This explicit computation is numerically unstable and is only possible for matrices Θ_H with small condition number. However, the matrix Θ_H is problem-specific and its condition number can be large. For large condition numbers, the inverse Θ_H^{-1} should only be computed implicitly, for example via an LU or QR decomposition. For this purpose, we write the transformation (4.29) equivalently as linear system

$$[\Theta_H \otimes \mathbf{I}_{\binom{n+1}{2}}] \hat{\mathbf{O}}^T = \check{\mathbf{O}}^T, \quad (4.30)$$

for $\hat{\mathbf{O}}^T$. This linear system is decoupled and can be written as

$$\Theta_H \hat{o}_{ij} = \check{o}_{ij}, \quad i = 1, \dots, n, \quad j = 1, \dots, \binom{n+1}{2} \quad (4.31)$$

with

$$\hat{o}_{ij} = \begin{bmatrix} \hat{\mathbf{H}}_{ij}^{(1)} \\ \vdots \\ \hat{\mathbf{H}}_{ij}^{(q_H)} \end{bmatrix} \quad \text{and} \quad \check{o}_{ij} = \begin{bmatrix} \check{\mathbf{H}}_{ij}^{(1)} \\ \vdots \\ \check{\mathbf{H}}_{ij}^{(q_H)} \end{bmatrix}. \quad (4.32)$$

The linear system (4.31) can be solved using standard linear algebra libraries that choose suitable solvers themselves, so these choices do not need to be made manually.

This decoupled approach is advantageous in terms of computational costs, parallelizability and numerical stability. Instead of solving the linear system (4.24) of dimension $sK \times q(n) = q_H \binom{n+1}{2} \times q_H \binom{n+1}{2}$ for the n columns of $\hat{\mathbf{O}}^T$, we hence solve the q_H linear systems (4.28) of dimension $\binom{n+1}{2} \times \binom{n+1}{2}$ for the n columns of $[\check{\mathbf{H}}^{(i)}]^T$ and the $n \binom{n+1}{2}$ linear systems (4.31) of dimension $q_H \times q_H$ for the vectors $\hat{o}_{ij}$. Table 4.1 summarizes the computational costs of solving these three linear systems by first computing the inverse of the respective system matrix – explicitly or implicitly – and then computing the individual solutions by matrix-vector-products of the inverse and the respective right-hand-side. As can be seen, the computational costs reduce significantly from $\mathcal{O}(n^6 q_H^3 + n^5 q_H^2)$ for the monolithic approach to $\mathcal{O}(n^6 + n^5 q_H + n^3 q_H^2 + q_H^3)$ for the decoupled approach. While the monolithic system (4.24) can be solved in parallel for the n columns of $\hat{\mathbf{O}}^T$, the decoupled approach can be split into even more parallel jobs in two steps. First, (4.28) can be solved in parallel for the $q_H n$ columns of the operators $[\check{\mathbf{H}}^{(i)}]^T$, $i = 1, \dots, q_H$. Second, (4.31) can be solved in parallel for the $n \binom{n+1}{2}$

Table 4.1: Summary of computational costs of solving the linear systems involved in the monolithic and decoupled approach.

	system dimension	number of solves	costs matrix inversion	costs individual solutions	total costs
monolithic (4.24)	$q_H \binom{n+1}{2} \times q_H \binom{n+1}{2}$	n	$\mathcal{O}((q_H n^2)^3)$	$n \cdot \mathcal{O}((q_H n^2)^2)$	$\mathcal{O}(n^6 q_H^3 + n^5 q_H^2)$
decoupled (4.28)	$\binom{n+1}{2} \times \binom{n+1}{2}$	$q_H n$	$\mathcal{O}((n^2)^3)$	$q_H n \cdot \mathcal{O}((n^2)^2)$	$\mathcal{O}(n^6 + n^5 q_H)$
decoupled (4.29)	$q_H \times q_H$	$n \binom{n+1}{2}$	$\mathcal{O}(q_H^3)$	$n \binom{n+1}{2} \cdot \mathcal{O}(q_H^2)$	$\mathcal{O}(n^3 q_H^2 + q_H^3)$

vectors $\hat{o}_{ij}$. In terms of numerical stability, the matrix $\mathbf{D} = \Theta_H \otimes \hat{\mathbf{X}}_2^T$ in the monolithic linear system (4.24) has condition number $\text{cond}(\mathbf{D}) = \text{cond}(\Theta_H) \text{cond}(\hat{\mathbf{X}}_2^T)$, whereas the decoupled approach involves the condition numbers $\text{cond}(\hat{\mathbf{X}}_2^T)$ and $\text{cond}(\Theta_H)$ separately in the linear systems (4.28) and (4.31), respectively.

As another benefit, the decoupling enables the application of methods developed for non-parametric operator inference. For example, the decoupling approach proposed in [6] can be applied to the decoupled problems (4.28) to reduce the computational costs and condition numbers even more.

4.5 ALTERNATIVE DERIVATION VIA LAGRANGE INTERPOLATION

An alternative derivation of the same decoupling approach is based on Lagrange interpolation. Concretely, [100] states that if the matrix Θ_H defined in (4.20) has full rank for the parameter samples μ_i , $i = 1, \dots, s$ and $s = q_H$, then we can construct alternative coefficient functions $\bar{\theta}_H^{(p)}$ such that

$$\sum_{p=1}^{q_H} \bar{\theta}_H^{(p)}(\mu) \bar{\mathbf{H}}^{(p)} = \sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \hat{\mathbf{H}}^{(p)}, \quad (4.33)$$

that satisfy the Lagrange property

$$\bar{\theta}_H^{(p)}(\mu_i) = \delta_{ip}, \quad \text{for all } i, p = 1, \dots, q_H, \quad (4.34)$$

for the given parameter samples μ_i , with the Kronecker delta δ_{ip} .

This alternative affine decomposition (4.33) allows trivial decoupling of (4.14) since $\bar{\Theta}_H = \mathbf{I}_s$, where $[\bar{\Theta}_H]_{ij} = \bar{\theta}_H^{(j)}(\mu_i)$. However, the result of this decoupled approach are the operators $\bar{\mathbf{H}}^{(i)}$, not the original ROM operators $\hat{\mathbf{H}}^{(i)}$. To compute these original operators, we combine (4.33) and (4.34) to find

$$\bar{\mathbf{H}}^{(i)} = \sum_{p=1}^{q_H} \theta_H^{(p)}(\mu_i) \hat{\mathbf{H}}^{(p)}, \quad \text{for all } i = 1, \dots, q_H, \quad (4.35)$$

and can then solve for $\widehat{\mathbf{H}}^{(p)}$.

Comparing the operator transformations (4.35) and (4.29), we realize that the operators $\bar{\mathbf{H}}^{(i)}$ coincide with the operators $\check{\mathbf{H}}^{(i)}$. Hence, both derivations lead to exactly the same computations. Note in particular that we do not need to compute the alternative coefficient functions $\bar{\theta}$ explicitly.

This Lagrange interpolation-based derivation does not require the snapshots $\widehat{\mathbf{X}}(\mu_i)$ to be parameter-independent. Future work should investigate whether this derivation allows decoupling also for parameter-dependent snapshots.

Note that the function $\sum_{p=1}^{q_H} \bar{\theta}_H^{(p)}(\mu) \bar{\mathbf{H}}^{(p)}$ interpolates the function values of the affine decomposition $\sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \widehat{\mathbf{H}}^{(p)}$ but not necessarily those of the true parameter-dependent operator (4.4) if $\sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \widehat{\mathbf{H}}^{(p)}$ is an approximation of (4.4).

4.6 NUMERICAL EXPERIMENTS

We consider the heat equation

$$\frac{\partial}{\partial t} T(\xi, t; \mu) = \frac{\partial}{\partial \xi} \left(\nu(T, \xi, \mu) \frac{\partial}{\partial \xi} T(\xi, t; \mu) \right) \quad (4.36)$$

with spatial coordinate $\xi \in \Omega = [0, 2\pi]$, time $t \in [0, 0.8]$, temperature $T : \Omega \times [0, 0.8] \rightarrow \mathbb{R}$ and two different thermal conductivities $\nu : \mathbb{R} \times \Omega \times \mathcal{D} \rightarrow \mathbb{R}^+$ defined in Sections 4.6.1 and 4.6.2. Both conductivities have a linear temperature dependency, so the corresponding spatially-discretized FOM has the form (4.1) with quadratic right-hand side (4.2). The conductivity in Section 4.6.1 implies an affine decomposition (4.5) of the FOM operator, whereas the conductivity that will be chosen in Section 4.6.2 allows such a decomposition only approximately.

In both cases, we impose homogeneous Dirichlet boundary conditions

$$T(0, t) = T(2\pi, t) = 0, \quad t \in [0, 0.8], \quad (4.37)$$

and the initial condition

$$T(\xi, 0) = \exp(-(\xi - \pi)^2) \sin\left(\frac{\xi}{2}\right), \quad \xi \in \Omega. \quad (4.38)$$

We discretize with linear finite elements on an equidistant grid in space with mesh width $\Delta\xi = 2^{-7}$, and with explicit Euler in time with time step size $\Delta t = 10^{-4}$, to obtain the time-discrete dynamical system of dimension $N = 127$ for the discretized temperature x_k ,

$$\begin{aligned} x_{k+1}(\mu) &= x_k(\mu) + \Delta t \mathbf{H}(\mu) x_k^2(\mu), \\ k &= 0, \dots, K-1, \quad K = 8000. \end{aligned} \quad (4.39)$$

We simulate this system for both conductivities twice: first to generate snapshots to compute POD bases, then to compute snapshots for exact operator inference. The accuracy of the inferred operators is then evaluated based on the relative error with respect to the intrusive operators,

$$\frac{\left\| \sum_{p=1}^{q_H} \theta_H^{(p)}(\boldsymbol{\mu}) \widehat{\mathbf{H}}^{(p)} - \tilde{\mathbf{H}}(\boldsymbol{\mu}) \right\|_F}{\left\| \tilde{\mathbf{H}}(\boldsymbol{\mu}) \right\|_F}, \quad (4.40)$$

for specified parameter samples $\boldsymbol{\mu}$. Errors with respect to the FOM are reported in Appendix C.2. The code is available on <https://github.com/h3rror/exactParamOpInf>.

4.6.1 Heat equation with affine conductivity

Inspired by [113], we consider a piecewise-defined thermal conductivity but with linear temperature dependency,

$$v(T, \zeta, \boldsymbol{\mu}) = \begin{cases} \mu_1 T & \text{for } \zeta \in [0, \frac{2\pi}{16}) =: \Omega_1 \\ \vdots & \\ \mu_i T & \text{for } \zeta \in [\frac{i-1}{16}2\pi, \frac{i}{16}2\pi) =: \Omega_i \\ \vdots & \\ \mu_{16} T & \text{for } \zeta \in [\frac{15}{16}2\pi, 2\pi] =: \Omega_{16} \end{cases} \quad (4.41)$$

with parameter $\boldsymbol{\mu} \in [1, 2]^{16}$. This conductivity allows an affine decomposition of the quadratic operator,

$$\mathbf{H}(\boldsymbol{\mu}) = \sum_{p=1}^{16} \mu_p \mathbf{H}^{(p)}, \quad (4.42)$$

where the terms $\mathbf{H}^{(p)}$, $p = 1, \dots, 16$, correspond to the conductivity 1 in Ω_p and 0 everywhere else, respectively.

We first simulate the FOM (4.39) for 16 parameter samples $\bar{\boldsymbol{\mu}}_i$, $i = 1, \dots, 16$ whose entries are defined as $[\bar{\boldsymbol{\mu}}_i] = 1 + \delta_{ij}$ with the Kronecker delta δ_{ij} for all $i, j = 1, \dots, 16$. The resulting $16 \cdot (8000 + 1)$ snapshots are used to compute the global POD basis $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ with $n = 1, \dots, 30$. Then we use the POD basis $\mathbf{V}_{30}$ to generate snapshot data for exact operator inference according to Algorithm 4.2 for the same 16 parameter samples $\boldsymbol{\mu}_i = \bar{\boldsymbol{\mu}}_i$, $i = 1, \dots, 16$. This choice of samples ensures that the matrix $\boldsymbol{\Theta}_H$ has full rank, so the solution to (4.14) is unique as discussed in Section 4.3.

Table 4.2: Number of linear systems of different dimensions solved during parametric operator inference for ROM dimension $n = 30$ and $q_H = 16$.

	$q_H \binom{n+1}{2} \times q_H \binom{n+1}{2}$ $= 7,440 \times 7,440$	$\binom{n+1}{2} \times \binom{n+1}{2}$ $= 465 \times 465$	$q_H \times q_H$ $= 16 \times 16$
monolithic	$n = 30$	o	o
decoupled	o	$\binom{n+1}{2} = 480$	$n \binom{n+1}{2} = 13,950$

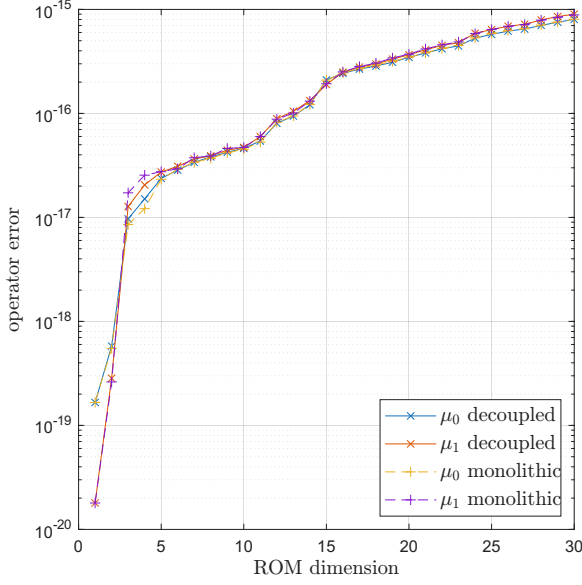
Fig. 4.1a shows the errors between the inferred and the intrusive operators evaluated at μ_1 , which is one of the parameter samples used in training, and μ_0 , whose entries are sampled randomly from the uniform distribution in $[1, 2]$ and which is not part of the parameter samples used in training. As can be seen, the errors are for both parameter values on the order of at most 10^{-15} , so our proposed method indeed reconstructs the intrusive operators up to machine precision. Furthermore, we cannot see any difference between the operators of the monolithic and our proposed decoupled approach. This confirms the correctness of the proposed decoupling approach. As described in Section 4.4, the decoupled approach has better parallelizability and significantly smaller costs. This is confirmed by Table 4.2, which shows the number of linear systems solved for this test case with ROM dimension $n = 30$ and $q_H = 16$. While the monolithic approach solves few very large linear systems, the decoupled approach solves many smaller systems. Consequently, the decoupled approach is much faster because the cost of solving linear systems scales cubically with their dimension. Additionally, Fig. 4.1b shows that the condition numbers of the matrices involved in the decoupled approach are smaller than the condition number of the monolithic matrix $\mathbf{D}$. Since Θ_H is independent of the ROM dimension, $\text{cond}(\Theta_H)$ is constant. As a result, the factor by which $\text{cond}(\hat{\mathbf{X}}_2^T)$ is smaller than $\text{cond}(\mathbf{D})$ is also constant.

4.6.2 Heat equation with non-affine conductivity

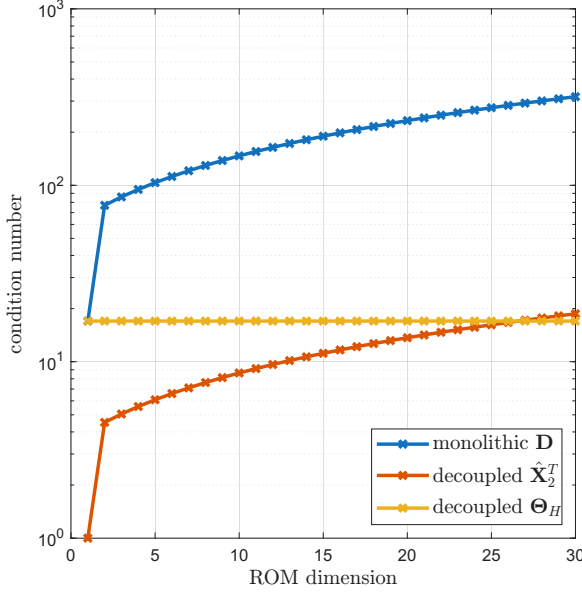
For the second test case consider, we consider the thermal conductivity

$$\nu(\xi, \mu, T) = (e^{-\frac{1}{2\pi}(\xi - \mu)^2} + 1)T, \quad (4.43)$$

with parameter $\mu \in \Omega$. This conductivity does not allow an exact affine decomposition of the quadratic operator. To obtain an approximate



(a) Relative combined operator errors (4.40) evaluated at μ_0 and μ_1 .



(b) Condition number of $\mathbf{D}$ for monolithic operator inference and $\hat{\mathbf{X}}_2^T$ and Θ_H for decoupled operator inference.

Figure 4.1: Results for the test case with affine conductivity (4.41).

affine decomposition (4.5) we use a Taylor expansion of the operator $\mathbf{H}(\mu)$ with $q_H = 16$ terms,

$$\mathbf{H}(\mu) \approx \sum_{p=0}^{q_H-1} \frac{(\mu - \mu_0)^p}{p!} \mathbf{H}^{(p)}. \quad (4.44)$$

The expansion point μ_0 can be chosen arbitrarily, here we choose $\mu_0 = 1.3\pi$.

We first simulate the FOM (4.39) for 5 equidistant parameter samples in Ω , $\bar{\mu}_i = 2\pi \frac{i-1}{5-1}$, $i = 1, \dots, 5$. The resulting $5 \cdot (8000 + 1)$ snapshots are used to compute the global POD bases $\mathbf{V}_n \in \mathbb{R}^{N \times n}$ with $n = 1, \dots, 30$. Then we use the POD basis $\mathbf{V}_{30}$ to generate snapshot data for exact operator inference according to Algorithm 4.2 for the $q_H = 16$ equidistant parameter samples in Ω , $\mu_i = 2\pi \frac{i-1}{16-1}$, $i = 1, \dots, 16$. For this choice of samples, the matrix Θ_H is the Vandermonde matrix of dimension 16 and has full rank, so the solution to (4.14) is unique as discussed in Section 4.3.

Fig. 4.2a shows the errors between the inferred and the intrusive operators evaluated both at the expansion point μ_0 and at $\mu_1 = 0$, which is the point in Ω with the largest distance to μ_0 . As in Fig. 4.1a, the errors of the decoupled and the monolithic approach do not deviate significantly from each other, so they confirm the correctness of the decoupling approach. However, compared to the previous test case depicted in Fig. 4.1a, the errors for μ_0 are larger by a few orders of magnitude. This error is not associated to the accuracy of the exact operator inference approach but due to the Taylor approximation (4.44) used to obtain an affine decomposition. This approximation causes an inconsistency because, on the one hand, operator inference fits the $q_H = 16$ basis operators $\hat{\mathbf{H}}^{(p)}$, $p = 1, \dots, 16$ in (4.14) with $\theta_H^{(p)}$ defined by the Taylor approximation (4.44), on the other hand, the snapshot data is generated from the true FOM operator $\mathbf{H}(\mu)$. This inconsistency is demonstrated in Fig. 4.2a by the error between the intrusive ROM operators $\tilde{\mathbf{H}}^{(p)}$ corresponding to the Taylor approximation (4.44) computed via (4.6) and the true ROM operator $\tilde{\mathbf{H}}(\mu)$ computed via (4.4),

$$\frac{\left\| \sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \tilde{\mathbf{H}}^{(p)} - \tilde{\mathbf{H}}(\mu) \right\|_F}{\left\| \tilde{\mathbf{H}}(\mu) \right\|_F}. \quad (4.45)$$

As can be seen, this error ranges from 10^{-18} to 10^{-15} for all ROM dimension when evaluated at the expansion point μ_0 . Indeed, the Taylor approximation (4.44) coincides with the true operator $\mathbf{H}(\mu)$ for this parameter value. For μ_1 , however, the error ranges from 10^{-8} to 10^{-2} . Since both the true ROM operator $\tilde{\mathbf{H}}(\mu)$ and the intrusive Taylor-based ROM operators $\tilde{\mathbf{H}}^{(p)}$ are computed intrusively via projection, this error

is purely due to the approximation error in the parameter space. Operator inference, on the other hand, implicitly learns a correction term $\Delta \mathbf{H}^{(p)}$ to the intrusive Taylor term, $\widehat{\mathbf{H}}^{(p)} = \tilde{\mathbf{H}}^{(p)} + \Delta \mathbf{H}^{(p)}$. As can be seen, the errors of the inferred operators evaluated at μ_1 range from 10^{-20} to 10^{-15} , similarly to the errors in Fig. 4.1a where no Taylor approximation was employed. The implicit correction seems to compensate the Taylor approximation error completely. For μ_0 , in contrast, the implicit correction even deteriorates the accuracy compared to the Taylor-based ROM operators with errors ranging from 10^{-15} to 10^{-9} .

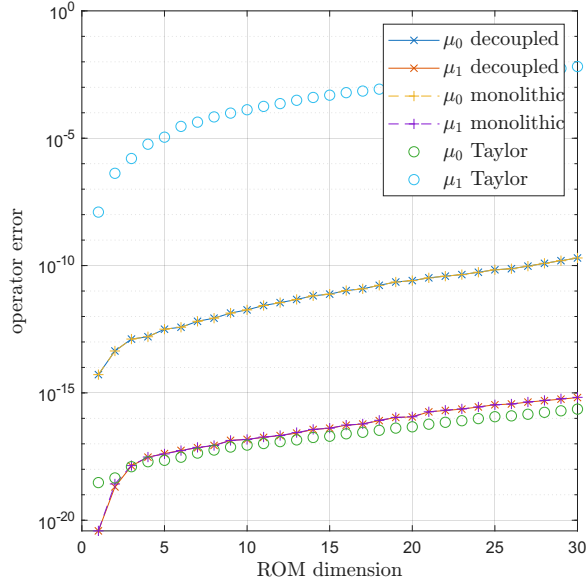
Fig. 4.2b shows again how the decoupling results in smaller conditions numbers. However, since the condition number of the Vandermonde matrix Θ_H itself is very large, the benefit is relatively small here. In fact, the large condition numbers of Vandermonde matrices is limiting the choice of q_H in this test case for both monolithic and decoupled approach.

4.7 CONCLUSION

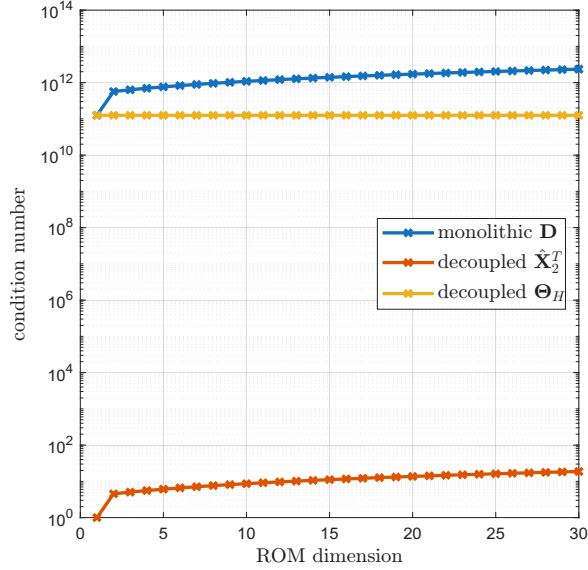
In this chapter, we have resolved three issues of parametric operator inference by extending our previous work on exact non-parametric operator inference in Chapter 3, in which smartly chosen initial conditions lead to snapshot data that guarantee a full rank inference problem. First, we can guarantee the uniqueness of the operator inference solution, provided that suitable parameter samples can be found. Second, we avoid the introduction of time discretization inconsistencies and non-Markovian information that could cause errors in the inferred operators. Third, we significantly reduce the computational costs by decoupling the minimization problem.

This decoupling exploits the exact operator inference snapshot data in two steps. First, the minimization problem simplifies to a linear system. Second, this high-dimensional monolithic linear system decouples into smaller linear systems. The resulting reduction in computational costs can be decisive for the total costs of operator inference, depending on the polynomial degree and the complexity of the parameter dependency of the underlying problem and the considered ROM dimension. Furthermore, the smaller linear systems can be solved embarrassingly parallel and have a smaller condition number and are hence numerically more stable.

In numerical experiments, we demonstrate that our proposed method reconstructs intrusive parametric ROM operators up to machine precision. The experiments confirm the correctness of the decoupling approach and exemplify the improvement in condition number when replacing the monolithic system by the decoupled systems.



(a) Relative combined operator errors (4.40) and (4.45) evaluated at $\mu_0 = 1.3\pi$ and $\mu_1 = 0$.



(b) Condition number of $\mathbf{D}$ for monolithic operator inference and $\hat{\mathbf{X}}_2$ and Θ_H for decoupled operator inference.

Figure 4.2: Results for the test case with non-affine conductivity (4.43).

Future work should investigate whether the presented Lagrange interpolation-based derivation is also applicable to parameter-dependent snapshot data, as well as extensions to dynamical systems that are derived from systems of partial differential equations such as the Navier-Stokes equations.

CONCLUSIONS

5.1 CONCLUSIONS

In this thesis, three advances of projection-based model reduction have been presented in the directions of structure-preserving, non-intrusive and parametric model reduction. While these advances address different problems, they have in common that they all achieve structure preservation. While Chapter 2 preserves structure of the FOM explicitly in the intrusive ROM, Chapters 3 and 4 preserve structures of the intrusive ROM in the non-intrusive ROM as a consequence of exact reconstruction.

Another commonality of the three contributions is that they all are accompanied with improvements in computational efficiency. The main motivation for the development of the ROM presented in Chapter 2 is to preserve the structure of the kinetic energy evolution in presence of time-dependent boundary conditions. This structure preservation is achieved thanks to an orthogonal decomposition of the velocity field that additionally enables the computation of the velocity without the computation of the pressure.

Chapter 3 presents a snapshot data generation method that guarantees the exact reconstruction of the intrusive ROM operators. This exactness is maintained throughout the method and leads to three improvements in computational efficiency: First, while existing works have generated large amounts of snapshot data following the heuristic that more data increases the probability for full rank, the exact control over which snapshots are generated enables the minimization of the amount of data. Second, the exactness of the snapshot data allows to replace the regression problem (least squares) by an interpolation problem (linear system). Third, the exact reconstruction implies that properties of the intrusive ROM are preserved in the inferred ROM automatically that otherwise have to be enforced, potentially at additional costs.

In Chapter 4, the full rank-ensuring snapshot data generation method is extended to parametric problems. As a by-product, the parametric operator inference problem simplifies such that it can be decoupled into smaller subproblems that are embarrassingly parallel, numerically more stable and can be solved more efficiently. The achieved improvement in computational efficiency in this chapter is in line with the intrusive ROM whose operators can be computed in the same decoupled way.

5.2 RECOMMENDATIONS

In future work, Chapters 2 and 3 should be combined to reconstruct the intrusive energy-consistent velocity-only ROM for the incompressible Navier-Stokes equations non-intrusively via operator inference. The inference of the operators of the velocity-only ROM ODE itself can be done straightforwardly. However, computing the ROM basis Φ_{hom} non-intrusively is not trivial since this basis is defined via POD of velocity snapshots after subtraction of the lifting function. How this lifting function can be constructed non-intrusively is subject to future research.

Combination of such research with Chapter 4 could be interesting as well, but is not necessary for all types of parametric studies. For example, varying inflow profiles can be expressed via different input signals without changing the ROM operators, and varying Reynolds numbers can be handled by rescaling the ROM diffusion operator. On the other hand, Chapter 4 can for example be useful for parametrized geometries.

When applying the energy-consistent ROM in practice, several aspects should be taken into account. As described in [11], not all singular value decomposition algorithms preserve the divergence-freeness of the POD basis. For complicated geometries, the extension to non-cartesian structure-preserving grids [108] should be investigated. Since DNS simulations are extremely expensive and can result in snapshots with a slow singular value decay, it should be investigated whether LES can be used as FOM instead. To maintain the structure preservation of the ROM, the involved turbulence model should be structure-preserving as well, in particular regarding the divergence-freeness [2]. For computational efficiency, energy-conserving hyper-reduction should be applied in intrusive settings [61].

In non-intrusive settings, hyper-reduction is not possible, as stated in Chapter 4, since the assumption of efficient access to individual entries of the FOM operator does not hold. In non-intrusive settings, it is therefore recommended to first use exact operator inference to reconstruct the ROM operators and then find a more efficient approximation of these operators in a second steps. In contrast to conventional hyper-reduction, this approximation does not act on the FOM operator but on the ROM operator, so correlations between FOM operator entries cannot be exploited in this approximation.

Since exact operator inference is explicitly developed for use in industry, there are only a few issues that could impede its use in practice. The only assumptions on the FOM solver are that snapshots can be generated with the prescribed full rank-ensuring initial conditions and explicit Euler as time discretization.

While prescribing initial conditions could be limited in different ways by specific software, the proposed set of full-rank ensuring initial conditions is by no means unique. For example, the interpretation of these initial conditions as interpolation nodes reveals that we can apply any invertible linear mapping to these nodes without violating the uniqueness of the corresponding interpolant. Furthermore, the constructive proof of uniqueness in Appendix B.1 motivates that alternative interpolation nodes, for example Chebychev points [107], can be chosen separately for each homogeneous multivariate polynomial that corresponds to a single ROM operator of certain polynomial degree. Hence, there are various opportunities to develop alternative initial condition sets in view of the limitations of a specific FOM solver.

The assumption of explicit Euler as time discretization can be a substantial limitation to exact reconstruction. While time discretization inconsistencies can be eliminated for other explicit Runge-Kutta schemes by reformulating these schemes as explicit Euler schemes as described in [77], the introduction of non-Markovian information is unavoidable. Future work should investigate how large the effect of such non-Markovian information is. Such error investigations should also be performed for settings that do not allow any generation of additional snapshot data.

If the FOM solver assumptions are satisfied, the only difference between the non-intrusive computation of the ROM operators via exact operator inference and the intrusive computation via multiplications of the FOM operators with the ROM basis is that exact operator inference involves solving a linear system. Solving this linear system could be expensive and introduce significant numerical errors if the system has large dimension and a high condition number. However, these problems can be addressed by decoupling the linear system as described in [6]. A complementary decoupling follows from the constructive proof in Appendix B.1. Lower condition numbers could be achieved for different snapshot sets, for example based on interpolation with Chebychev points [107].

A more conceptual assumption is the knowledge of the polynomial structure of the ROM right-hand side. While we presume that the polynomial degree can be inferred from the underlying PDE, additional lower-order polynomials could for example be introduced due to the discretization of boundary conditions. In such cases, it is recommended to include all possible terms in the operator inference as done in Section 3.4.1 where a quadratic operator is inferred although not present in the ROM right-hand side. To verify whether the inferred model is complete, one can perform exact operator inference on different snapshot data, for example by rotating the full-rank ensuring ROM states using an orthogonal matrix. In case of a complete model, the result should be

the same, in case of an incomplete model, the inferred operators can be different.

5.3 BROADER OUTLOOK

Within this thesis, we have only considered POD-based model reduction. A well-known limitation of POD is that advection-dominated processes cannot be approximated well within low-dimensional linear subspaces, the slow decay of the Kolmogorov- n -width. Approaches to mitigate this limitation include nonlinear model reduction and adaptive model reduction [55]. Another modification of the POD-ROM is replacing the Galerkin projection by a Petrov-Galerkin projection to improve stability [23] and accuracy of the reduced dynamics [75]. Whether these modifications are compatible with the methods presented in this thesis is subject to future research.

Beyond projection-based model reduction, the interpretation of data samples as interpolation nodes can be useful for other problems where model parameters are fitted to data such as data assimilation or SINDy [18]. The interpolation interpretation also holds for non-polynomial models, but the identification of interpolation nodes that define a unique interpolant can be more difficult.

SUPPLEMENTARY MATERIAL FOR CHAPTER 2

A.1 IMPORTANCE OF ORTHOGONAL LIFTING FUNCTION

To comprehend the importance of the orthogonality between the lifting function $\tilde{V}_{\text{inhom}}(t)$ and the homogeneous velocity basis Φ_{hom} (2.27), we consider an alternative pair of a lifting function $\hat{V}_{\text{inhom}}(t)$ and a homogeneous velocity basis $\hat{\Phi}_{\text{hom}}$ that are not Ω_h -orthogonal.

For this purpose, let us assume that the lifting function $\hat{V}_{\text{inhom}}(t)$ satisfies the approximated mass equation

$$M_h \hat{V}_{\text{inhom}}(t) = \tilde{y}_M(t), \quad (\text{A.1})$$

but not the definition (2.26). We define the difference between the two lifting functions $\Delta(t) = \hat{V}_{\text{inhom}}(t) - \tilde{V}_{\text{inhom}}(t)$.

For simplicity, we assume that the approximated boundary condition vector equals the exact boundary condition vector, $\tilde{y}_{\text{bc}}(t) = y_{\text{bc}}(t)$ and hence $\tilde{V}_{\text{inhom}}(t) = V_{\text{inhom}}(t)$. Then, we can compute the homogeneous velocity basis $\hat{\Phi}_{\text{hom}}$ as the Ω_h -orthonormal POD basis of snapshots $\hat{V}_{h,\text{hom}}^j$ of the homogeneous velocity field $\hat{V}_{h,\text{hom}}(t) = V_h(t) - \hat{V}_{\text{inhom}}(t)$.

Then, $\hat{V}_{r,\text{hom}}(t) = \hat{\Phi}_{\text{hom}} \hat{a}_{\text{hom}}(t)$ approximates $\hat{V}_{h,\text{hom}}(t)$, where $\hat{a}_{\text{hom}}(t)$ is defined by replacing $V_h(t)$ in the momentum equation (2.5) by $\hat{V}_r(t) = \hat{V}_{r,\text{hom}}(t) + \hat{V}_{\text{inhom}}(t)$, and projecting onto $\hat{\Phi}_{\text{hom}}$,

$$\begin{aligned} \hat{\Phi}_{\text{hom}}^T \Omega_h \frac{d}{dt} (\hat{V}_{h,\text{hom}}(t) + \hat{V}_{\text{inhom}}(t)) \\ = \hat{\Phi}_{\text{hom}}^T F_h^{CD} (\hat{V}_r(t), \tilde{y}_{\text{bc}}(t)) - \hat{\Phi}_{\text{hom}}^T \mathbf{G}_h p_h(t). \end{aligned} \quad (\text{A.2})$$

By construction, we have $M_h \hat{\Phi}_{\text{hom}} = \mathbf{0}$, so the pressure term vanishes. However, we generally do not have orthogonality of components,

$$\hat{\Phi}_{\text{hom}}^T \Omega_h \hat{V}_{\text{inhom}}(t) = \hat{\Phi}_{\text{hom}}^T \Omega_h \Delta(t) =: \Delta_r(t). \quad (\text{A.3})$$

Hence, the evolution of $\hat{a}_{\text{hom}}(t)$ is given by

$$\frac{d}{dt} \hat{a}_{\text{hom}}(t) = \hat{\Phi}_{\text{hom}}^T F_h^{CD} (\hat{V}_r(t), \tilde{y}_{\text{bc}}(t)) - \frac{d}{dt} \Delta_r(t). \quad (\text{A.4})$$

The pressure $\hat{p}_r(t)$ corresponding to $\hat{V}_r(t)$ can be retrieved from

$$\bar{\mathbf{L}}_h \hat{p}_r(t) = -\frac{d}{dt} \tilde{y}_M(t) + \mathbf{M}_h \Omega_h^{-1} F_h^{CD} (\hat{V}_r(t), \tilde{y}_{\text{bc}}(t)). \quad (\text{A.5})$$

Then, the kinetic energy $\hat{K}_r(t) = \frac{1}{2} \|\hat{\mathbf{V}}_r\|_{\Omega_h}^2$ has the evolution

$$\frac{d}{dt} \hat{K}_r(t) = \hat{\mathbf{V}}_r(t) F_h^{CD}(\hat{\mathbf{V}}_r(t), \tilde{\mathbf{y}}_{bc}(t)) + \tilde{\mathbf{y}}_M(t)^T \hat{\mathbf{p}}_r(t) \quad (\text{A.6})$$

$$+ \left(\Delta_r(t)^T \hat{\Phi}_{\text{hom}}^T - \Delta^T \right) F_h^{CD}(\hat{\mathbf{V}}_r(t), \tilde{\mathbf{y}}_{bc}(t)) \quad (\text{A.7})$$

$$+ \Delta(t)^T \Omega_h \frac{d}{dt} \hat{\mathbf{V}}_{\text{inhom}}(t) \quad (\text{A.8})$$

$$+ \hat{\mathbf{V}}_{\text{inhom}}(t)^T \Omega_h \frac{d}{dt} \Delta(t) \quad (\text{A.9})$$

$$- \Delta(t)^T \Omega_h \frac{d}{dt} \Delta(t) - \Delta_r(t)^T \frac{d}{dt} \Delta_r(t). \quad (\text{A.10})$$

Hence, this kinetic energy evolution does not have the same structure as the energy evolutions (2.30) and (2.36), because of the additional terms in (A.7), (A.8) and (A.9). Note that the lifting function $\hat{\mathbf{V}}_{\text{inhom}}(t)$ defined by (2.26) is the only lifting function that satisfies the approximated mass equation (A.1) and is Ω_h -orthogonal to all homogeneous velocities, i.e.

$$\mathbf{V}^T \Omega_h \tilde{\mathbf{V}}_{\text{inhom}}(t) = 0 \quad \text{for all } \mathbf{M}_h \mathbf{V} = \mathbf{0}. \quad (\text{A.11})$$

Hence, any lifting function $\hat{\mathbf{V}}_{\text{inhom}}(t)$ that satisfies (A.1) and (A.11) implies $\Delta(t) = \mathbf{0}$, so the additional terms (A.7), (A.8), (A.9) and (A.10) vanish.

A.2 EFFECT OF TIME INTEGRATION METHOD: ENERGY-CONSERVING RUNGE-KUTTA METHODS

We investigate the effect of using an energy-conserving, implicit Runge-Kutta method [97], the implicit midpoint rule, instead of the explicit fourth-order Runge-Kutta method used in Section 2.6. As can be seen in Fig. A.1, the difference between the explicitly and implicitly integrated ROMs is quite small compared to the velocity error of the explicitly integrated ROMs depicted in Fig. 2.6.

As discussed in Section 2.4, energy-conserving Runge-Kutta methods do not conserve the kinetic energy for the considered test cases since the underlying ODEs conserve kinetic energy neither. In addition, Fig. A.2 shows that the relative kinetic energy errors do not vanish even when an energy-conserving Runge-Kutta method is used.

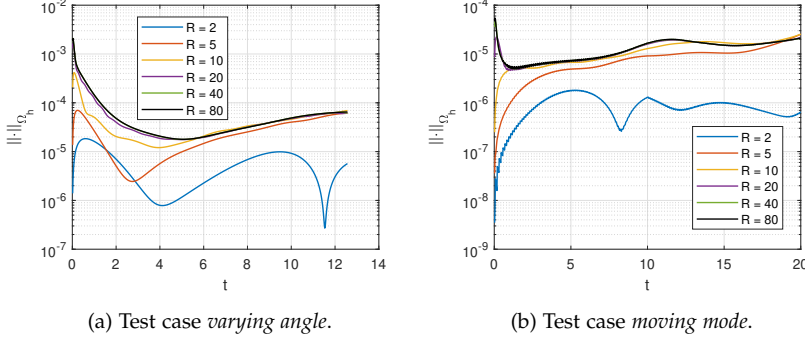


Figure A.1: Ω_h -norm of ROM velocities $\|V_{r,\text{explicit}}^j - V_{r,\text{implicit}}^j\|_{\Omega_h}$ integrated in time with explicit RK4 and with implicit midpoint for the two testcases with boundary condition best approximation and $R_{\text{hom}} = R_{\text{bc}} = R$.

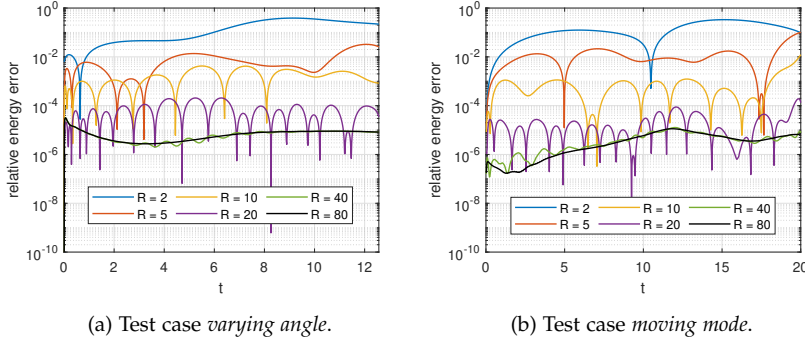


Figure A.2: Relative kinetic energy error $\frac{|K_h^j - K_h^j|}{\text{average}_j[K_h^j]}$ for both testcases, integrated in time with implicit midpoint with boundary condition best approximation and $R_{\text{hom}} = R_{\text{bc}} = R$.

A.3 BEST APPROXIMATION ERROR RELATIONS BETWEEN THE PROPOSED VELOCITY-ONLY ROM AND A STANDARD POD VELOCITY-PRESSURE ROM

We repeat and prove the statement in Section 2.5.3 about best approximation errors of standard POD bases and the proposed velocity-only ROM basis.

Proposition A.1 (Best approximation errors of velocity-only ROM and standard POD velocity-pressure ROMs). *Let, as in (2.50),*

$$\varepsilon(\boldsymbol{\phi}) = \sum_{j=1}^K \left\| \mathbf{V}_h^j - \boldsymbol{\phi} \boldsymbol{\phi}^T \boldsymbol{\Omega}_h \mathbf{V}_h^j \right\|_{\boldsymbol{\Omega}_h}^2, \quad (\text{A.12})$$

the best approximation error over a set of snapshots $\left\{ \mathbf{V}_h^j \right\}_{j=1}^K$. Then, the $\boldsymbol{\Omega}_h$ -orthonormal POD bases over the set of snapshots $\left\{ \mathbf{V}_h^j \right\}_{j=1}^K$ with $R_V = R_{\text{hom}}$ and $R_V = R_{\text{hom}} + R_{\text{in hom}}$ modes, $\boldsymbol{\Phi}_{[R_{\text{hom}}]}$ and $\boldsymbol{\Phi}_{[R_{\text{hom}} + R_{\text{in hom}}]}$, satisfy relation (2.51),

$$\varepsilon \left(\boldsymbol{\Phi}_{[R_{\text{hom}} + R_{\text{in hom}}]} \right) \leq \varepsilon \left([\boldsymbol{\Phi}_{\text{hom}} \quad \boldsymbol{\Phi}_{\text{in hom}}] \right) \leq \varepsilon \left(\boldsymbol{\Phi}_{[R_{\text{hom}}]} \right), \quad (\text{A.13})$$

with $R_{\text{in hom}}$, $\boldsymbol{\Phi}_{\text{hom}} \in \mathbb{R}^{N_V \times R_{\text{hom}}}$ and $\boldsymbol{\Phi}_{\text{in hom}} \in \mathbb{R}^{N_V \times R_{\text{in hom}}}$ as used in the proof of Proposition 2.1, and the assumption that $\tilde{\mathbf{y}}_{\text{bc}}(t) = \mathbf{y}_{\text{bc}}(t)$ for all $t \in [0, T]$.

Proof. We use the well-known optimality property of POD bases [114] that the $\boldsymbol{\Omega}_h$ -orthonormal POD basis $\boldsymbol{\Phi} \in \mathbb{R}^{N_V \times R_V}$ minimizes $\varepsilon(\boldsymbol{\phi})$ over all $\boldsymbol{\Omega}_h$ -orthonormal $\boldsymbol{\phi} \in \mathbb{R}^{N_V \times R_V}$. Since $\boldsymbol{\Phi}_{[R_{\text{hom}} + R_{\text{in hom}}]}$ and $[\boldsymbol{\Phi}_{\text{hom}} \quad \boldsymbol{\Phi}_{\text{in hom}}]$ both have the same dimension, this optimality property implies

$$\varepsilon \left(\boldsymbol{\Phi}_{[R_{\text{hom}} + R_{\text{in hom}}]} \right) \leq \varepsilon \left([\boldsymbol{\Phi}_{\text{hom}} \quad \boldsymbol{\Phi}_{\text{in hom}}] \right). \quad (\text{A.14})$$

To prove the second inequality, we use that

$$\boldsymbol{\Phi}_{\text{hom}}^T \boldsymbol{\Omega}_h \boldsymbol{\Phi}_{\text{in hom}} = \mathbf{0} \quad (\text{A.15})$$

and, thanks to the assumption that $\tilde{\mathbf{y}}_{\text{bc}}(t) = \mathbf{y}_{\text{bc}}(t)$ for all $t \in [0, T]$,

$$\mathbf{V}_{\text{in hom}}(t^j) \in \text{span}(\boldsymbol{\Phi}_{\text{in hom}}) \quad \text{for all } j = 1, \dots, K, \quad (\text{A.16})$$

where t^j is chosen such that $\mathbf{V}_h^j = \mathbf{V}_{h, \text{hom}}^j + \mathbf{V}_{\text{in hom}}(t^j)$. Then, we find

$$\boldsymbol{\Phi}_{\text{in hom}} \boldsymbol{\Phi}_{\text{in hom}}^T \boldsymbol{\Omega}_h \mathbf{V}_h^j = \mathbf{V}_{\text{in hom}}(t^j), \quad (\text{A.17})$$

and hence

$$\begin{aligned} V_h^j - [\Phi_{\text{hom}} \quad \Phi_{\text{inhom}}][\Phi_{\text{hom}} \quad \Phi_{\text{inhom}}]^T \Omega_h V_h^j \\ = V_{h,\text{hom}}^j - \Phi_{\text{hom}} \Phi_{\text{hom}}^T \Omega_h V_{h,\text{hom}}^j, \end{aligned} \quad (\text{A.18})$$

for all $j = 1, \dots, K$, and

$$\varepsilon([\Phi_{\text{hom}} \quad \Phi_{\text{inhom}}]) = \sum_{j=1}^K \left\| V_{h,\text{hom}}^j - \Phi_{\text{hom}} \Phi_{\text{hom}}^T \Omega_h V_{h,\text{hom}}^j \right\|_{\Omega_h}^2. \quad (\text{A.19})$$

Since Φ_{hom} is the Ω_h -orthonormal POD basis over the set of snapshots $\{V_{h,\text{hom}}^j\}_{j=1}^K$, the POD optimality property implies that Φ_{hom} minimizes $\varepsilon([\phi \quad \Phi_{\text{inhom}}])$ over all Ω_h -orthonormal $\phi \in \mathbb{R}^{N_V \times R_V}$. Hence, we find

$$\varepsilon([\Phi_{\text{hom}} \quad \Phi_{\text{inhom}}]) \leq \varepsilon([\Phi_{[R_{\text{hom}}]} \quad \Phi_{\text{inhom}}]). \quad (\text{A.20})$$

Furthermore, due to the monotonicity of ε upon adding modes, we find

$$\varepsilon([\Phi_{[R_{\text{hom}}]} \quad \Phi_{\text{inhom}}]) \leq \varepsilon(\Phi_{[R_{\text{hom}}]}), \quad (\text{A.21})$$

which completes the proposition statement. $\square$

SUPPLEMENTARY MATERIAL FOR CHAPTER 3

B.1 PROOF OF FULL RANK OF OPERATOR INFERENCE LEAST SQUARES MATRIX

We prove that the ROM states and input signals proposed in Section 3.3.1 guarantee that the matrix $\mathbf{P}$ has full rank. For this purpose, we first note that $\mathbf{P}$ is composed of the submatrices $\mathbf{P} \in \mathbb{R}^{n_p \times n_p}$, $\mathbf{Z} \in \mathbb{R}^{n_p \times N_u}$, $\mathbf{0} \in \mathbb{R}^{N_u \times n_p}$ and $\mathbf{U} \in \mathbb{R}^{N_u \times N_u}$,

$$\begin{aligned} \mathbf{P} &= \begin{bmatrix} p(\tilde{\mathbf{x}}_1, \mathbf{0}) & \dots & p(\tilde{\mathbf{x}}_{n_p}, \mathbf{0}) & p(\mathbf{0}, u_1) & \dots & p(\mathbf{0}, u_p) \end{bmatrix} \\ &= \begin{bmatrix} \bar{\mathbf{P}} & \mathbf{Z} \\ \mathbf{0} & \mathbf{U} \end{bmatrix}. \end{aligned} \quad (\text{B.1})$$

The matrix $\bar{\mathbf{P}} \in \mathbb{R}^{n_p \times n_p}$ is the interpolation matrix corresponding to the multivariate interpolation problem of finding the coefficients c_α of the polynomial

$$p : \mathbb{R}^n \rightarrow \mathbb{R}, \quad \tilde{\mathbf{x}} \mapsto \sum_{|\alpha| \in \mathcal{I}} c_\alpha \tilde{\mathbf{x}}^\alpha \quad (\text{B.2})$$

such that

$$p(\tilde{\mathbf{x}}_i) = b_i, \quad i = 1, \dots, n_p \quad (\text{B.3})$$

for the interpolation nodes $\tilde{\mathbf{x}}_i \in \cup_{i \in \mathcal{I}} \mathcal{X}^i$ and arbitrary given interpolation values $b_i \in \mathbb{R}$. All entries of $\mathbf{Z}$ are equal to $p(\mathbf{0})$. $\mathbf{U} \in \mathbb{R}^{N_u \times N_u}$ is the N_u -dimensional identity matrix.

Since $\mathbf{P}$ is a block-triangular matrix, it has full rank if its diagonal blocks $\bar{\mathbf{P}}$ and $\mathbf{U}$ have full rank. While the identity matrix $\mathbf{U}$ trivially has full rank, proving that the polynomial interpolation matrix $\bar{\mathbf{P}}$ has full rank is more involved.

Most works on multivariate polynomial interpolation consider the special case $\mathcal{I} = \mathbb{N}_l := \{0, 1, \dots, l\}$ for a given $l \in \mathbb{N}$ [39, 40]. In Chapter 3, however, we allow *gaps* in the degree set $\mathcal{I}$, i.e., not all elements of

$\mathbb{N}_l$ are necessarily contained in $\mathcal{I}$. Therefore, we define for all $n \in \mathbb{N}^+$, $l \in \mathbb{N}_0$ and any $\mathcal{I} \subset \mathbb{N}_l$, the set of *gappy polynomials* in $\mathcal{I}$,

$$\mathcal{P}_{\mathcal{I}}^n := \left\{ p : \mathbb{R}^n \rightarrow \mathbb{R}, \quad \tilde{\mathbf{x}} \mapsto \sum_{|\alpha| \in \mathcal{I}} c_{\alpha} \tilde{\mathbf{x}}^{\alpha} \right. \\ \left. \left| \quad c_{\alpha} \in \mathbb{R}, \alpha \in \{0, 1, \dots, l\}^n \right. \right\} \quad (\text{B.4})$$

Then, the following theorem implies that $\bar{\mathbf{P}}$ has full rank.

Theorem B.1 (Gappy multivariate polynomial interpolation). *For any $n \in \mathbb{N}^+$, $l \in \mathbb{N}_0$, for any degree set $\mathcal{I} \subset \mathbb{N}_l$, there exists a unique polynomial $p \in \mathcal{P}_{\mathcal{I}}^n$ such that*

$$p(\mathbf{q}_j) = b_j, \quad j = 1, \dots, n_p, \quad (\text{B.5})$$

for the interpolation nodes $\mathbf{q}_j$ in $\cup_{i \in \mathcal{I}} \mathcal{X}^i$ and arbitrary interpolation values b_j .

B.1.1 Outline of proof

The special case of degree sets without gaps, (i.e. $\mathcal{I} = \mathbb{N}_l$), is covered by an existing result on existence and uniqueness of an interpolating multivariate polynomial [73] described in Section B.1.2. In Section B.1.3, we use this result to prove Theorem B.1 for the special case that $\mathcal{I}$ contains only one element. I.e., we can find for all $i \in \mathcal{I}$ a homogeneous polynomial $p_i \in \mathcal{P}_{\{i\}}^n$ that interpolates in the nodes in $\mathcal{X}^i$. We use this result as base case in a proof by induction in Section B.1.5 to prove the general case where $\mathcal{I}$ contains multiple elements. There, the induction hypothesis states that Theorem B.1 holds for a fixed degree set $\mathcal{I}^-$, i.e., there exists a unique polynomial p^- that interpolates in the interpolation nodes in $\cup_{i \in \mathcal{I}^-} \mathcal{X}^i$. In the induction step, we show that Theorem B.1 also holds for the new degree set $\mathcal{I} := \{i^*\} \cup \mathcal{I}^-$ for any $i^* \in \mathbb{N}_0$ with $i^* < \min \mathcal{I}^-$. For this purpose, we add a new polynomial p that has two key properties. First, p attains suitable values in the new interpolation nodes in $\mathcal{X}^{i^*}$ such that the resulting polynomial $p^- + p$ attains the prescribed interpolation values b_j . Second, p attains 0 in the old interpolation nodes in $\cup_{i \in \mathcal{I}^-} \mathcal{X}^i$ to not interfere with the interpolation property of p^- in these nodes. In Section B.1.4, we show that such a polynomial p exists in $\mathcal{P}_{\mathcal{I}}^n$.

B.1.2 Multivariate polynomial interpolation without gaps

The following theorem is given in [73] for the special case without gaps in the degree set, so $\mathcal{I} = \mathbb{N}_l$.

Theorem B.2 (Multivariate polynomial interpolation without gaps). *For any $m \in \mathbb{N}_0$, let $X_0, X_1, \dots, X_m \in \mathbb{R}^m$ be the vertices of a non-degenerate m -dimensional simplex in $\mathbb{R}^m$. For any $l \in \mathbb{N}_0$, we define the lattice*

$$B(l, m) := \left\{ x \in \mathbb{R}^m \mid x = \sum_{i=0}^m \lambda_i X_i, \lambda_i \in \mathbb{N}_0, \sum_{i=0}^m \lambda_i = l \right\}. \quad (\text{B.6})$$

Then there exists a unique polynomial $p \in \mathcal{P}_{\mathbb{N}_l}^m$ such that

$$p(q_i) = f_i, \quad i = 1, 2, \dots, \binom{m+l}{m}, \quad (\text{B.7})$$

for the interpolation nodes q_i in $B(l, m)$ and arbitrary interpolation values $f_i \in \mathbb{R}$.

B.1.3 Homogeneous multivariate polynomial interpolation

Next we consider the special case that the degree set $\mathcal{I}$ contains only one element l , so the interpolant is a homogeneous polynomial in $\mathcal{P}_{\{l\}}^n$ and the interpolation nodes are given by $\mathcal{X}^l$.

Theorem B.3 (Homogeneous multivariate polynomial interpolation). *For any $n \in \mathbb{N}^+$, $l \in \mathbb{N}_0$, there exists a unique polynomial $p \in \mathcal{P}_{\{l\}}^n$ such that*

$$p(q_i) = f_i, \quad i = 1, 2, \dots, \binom{n+l-1}{l}, \quad (\text{B.8})$$

for the interpolation nodes q_i in $\mathcal{X}^l$ as defined in (3.14) and arbitrary interpolation values $f_i \in \mathbb{R}$.

To prove this theorem, we trace it back to Theorem B.2 on multivariate polynomial interpolation without gaps. First, we note that $\mathcal{X}^l$ lies in an $(n-1)$ -dimensional subspace of $\mathbb{R}^n$ and constitutes a lattice of degree l and dimension $n-1$. Hence, we can construct a multivariate polynomial $p \in \mathcal{P}_{\mathbb{N}_l}^n$ that interpolates arbitrary values in $\mathcal{X}^l$. Then, we use the linear dependence of the nodes in $\mathcal{X}^l$ to reformulate p into an equivalent polynomial in $\mathcal{P}_{\{l\}}^n$.

Proof of Theorem B.3. We employ Theorem B.2 with $m = n-1$ and the vertices $X_0 = \mathbf{0} \in \mathbb{R}^{n-1}$ and X_j the unit vectors in $\mathbb{R}^{n-1}$ for $j = 1, \dots, n-1$. The theorem states that there exists a polynomial

$$\hat{p} : \mathbb{R}^{n-1} \rightarrow \mathbb{R}, \quad \hat{p}(\hat{x}) = \sum_{|\hat{\alpha}| \leq l} \hat{a}_{\hat{\alpha}} \hat{x}^{\hat{\alpha}}, \quad (\text{B.9})$$

with $\hat{a}_{\hat{\mathbf{k}}} \in \mathbb{R}$ for all multi-exponents $\hat{\mathbf{k}} \in \{0, 1, \dots, l\}^{n-1}$ with $|\hat{\mathbf{k}}| \leq l$, such that

$$\hat{p}(\hat{q}_i) = f_i, \quad i = 1, 2, \dots, \binom{n-1+l}{n-1} \quad (\text{B.10})$$

for the prescribed interpolation values f_i and the interpolation nodes $\hat{q}_i$ in $B(l, n-1)$.

We define the linear map $\mathbf{m}$ that maps the vertices $\mathbf{X}_j$ to the unit vectors in $\mathbb{R}^n$. Concretely, $\mathbf{m}(\mathbf{X}_j) = \mathbf{e}_j$, $j = 1, \dots, n-1$ and $\mathbf{m}(\mathbf{X}_0) = \mathbf{e}_n$. In view of the definition of $\mathcal{X}^i$ (3.14), we can rewrite the definition of $B(l, m)$ (B.6),

$$B(l, m) = \left\{ \sum_{j=1}^l a_j \mid a_1, \dots, a_l \in \{\mathbf{X}_0, \dots, \mathbf{X}_m\} \right\}. \quad (\text{B.11})$$

Hence, we find

$$\begin{aligned} \mathbf{m}(B(l, n-1)) \\ = \left\{ \sum_{j=1}^l a_j \mid a_1, \dots, a_l \in \{\mathbf{m}(\mathbf{X}_0), \dots, \mathbf{m}(\mathbf{X}_{n-1})\} \right\} = \mathcal{X}^l. \end{aligned} \quad (\text{B.12})$$

Consequently, the polynomial

$$p(\mathbf{x}) := \hat{p}(\mathbf{m}^{-1}(\mathbf{x})) \quad (\text{B.13})$$

satisfies (B.8) with the q_i in $\mathcal{X}^l$.

To show that this polynomial p is an element of $\mathcal{P}_{\{l\}}^n$, we note that the map $\mathbf{m}$ is given by

$$\mathbf{m} : \mathbb{R}^{n-1} \rightarrow \mathbb{R}^n, \quad \hat{\mathbf{x}} \mapsto \begin{bmatrix} I_{n-1} \\ \mathbf{0}^T \end{bmatrix} \hat{\mathbf{x}} + \begin{bmatrix} \mathbf{0} \\ l \end{bmatrix}, \quad (\text{B.14})$$

with the inverse map

$$\mathbf{m}^{-1} : \mathbb{R}^n \rightarrow \mathbb{R}^{n-1}, \quad \mathbf{x} \mapsto [I_{n-1} \quad \mathbf{0}] \mathbf{x}. \quad (\text{B.15})$$

We apply the definition of $\mathbf{m}^{-1}$ to find

$$p(\mathbf{x}) = \hat{p}(\mathbf{m}^{-1}(\mathbf{x})) = \sum_{|\hat{\mathbf{k}}| \leq l} \hat{a}_{\hat{\mathbf{k}}} (\mathbf{m}^{-1}(\mathbf{x}))^{\hat{\mathbf{k}}} = \sum_{|\hat{\mathbf{k}}| \leq l} \hat{a}_{\hat{\mathbf{k}}} \prod_{i=1}^{n-1} x_i^{\hat{k}_i}, \quad (\text{B.16})$$

with the entries x_i of $\mathbf{x} \in \mathbb{R}^n$, $i = 1, \dots, n$. By definition, all $\mathbf{x} \in \mathcal{X}^l$ satisfy

$$\sum_{i=1}^n x_i = l, \quad (\text{B.17})$$

so we can multiply $1 = \frac{1}{l} \sum_{i=1}^n x_i$ to get

$$p(\mathbf{x}) = \sum_{|\hat{\mathbf{a}}| \leq l} \hat{a}_{\hat{\mathbf{a}}} \left(\prod_{i=1}^{n-1} x_i^{\hat{a}_i} \right) \left(\frac{1}{l} \sum_{i=1}^n x_i \right)^{l - \sum_{i=1}^n \hat{a}_i}. \quad (\text{B.18})$$

As a result, each summand is a monomial of degree l , so we can write

$$p(\mathbf{x}) = \sum_{|\alpha|=l} a_{\alpha} \mathbf{x}^{\alpha} \quad (\text{B.19})$$

with suitable $a_{\alpha} \in \mathbb{R}$, for all $\alpha \in \{0, 1, \dots, l\}^n$ with $|\alpha| = l$. Thereby, we have proven existence of a polynomial $p \in \mathcal{P}_{\{l\}}^n$ that satisfies (B.8) for any interpolation values $f_i, i = 1, 2, \dots, \binom{n+l-1}{l}$. Since the corresponding interpolation matrix is square, existence of a solution implies also uniqueness. $\square$

B.1.4 Specific gappy multivariate polynomial interpolation

As explained in Section B.1.1, the proof of Theorem B.1 in Section B.1.5 involves a polynomial $p \in \mathcal{P}_{\mathcal{I}}^n$ with $\mathcal{I} := \{i^*\} \cup \mathcal{I}^-$ such that

$$p(\mathbf{q}_j) = 0, \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^i \text{ for all } i \in \mathcal{I}^- \quad (\text{B.20})$$

$$p(\mathbf{q}_j) = f_j, \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^{i^*}, \quad (\text{B.21})$$

for prescribed interpolation values $f_j \in \mathbb{R}$. To show that such a polynomial p exists, we make the Ansatz

$$p(\mathbf{q}) = p_{i^*}(\mathbf{q}) \hat{p} \left(\sum_{k=1}^n q_k \right) \quad \forall \mathbf{q} \in \mathbb{R}^n, \quad (\text{B.22})$$

with $p_{i^*} \in \mathcal{P}_{i^*}^n$ that satisfies (B.21) according to Theorem B.3, and a univariate polynomial $\hat{p}$ that solves the specific interpolation problem with interpolation values 0 and 1 described in the following lemma.

Lemma B.4 (Specific gappy univariate polynomial interpolation). *For any $l \in \mathbb{N}_0$, for any $\mathcal{I} := \{i^*\} \cup \mathcal{I}^- \subset \mathbb{N}_l$ with $i^* < \min \mathcal{I}^-$, there exists a polynomial $p \in \mathcal{P}_{\mathcal{I}^- i^*}^1$ with $\mathcal{I}^- i^* := \{i - i^* | i \in \mathcal{I}\}$ such that*

$$p(i) = 0, \quad \text{for all } i \in \mathcal{I}^- \text{ and} \quad (\text{B.23})$$

$$p(i^*) = 1. \quad (\text{B.24})$$

To prove this lemma, we show in Lemma B.5 that for fixed $l \in \mathbb{N}_0$, the existence of solutions to such specific interpolation problems with interpolation values 0 and 1 as described above implies the existence

of solutions to the general interpolation problems with arbitrary interpolation values. We use this result to prove Lemma B.4 with a case discrimination: In Lemma B.6, we consider the case $i^* \neq 0$ and in Lemma B.7 the case $i^* = 0$.

Lemma B.5 (Specific and general gappy univariate interpolation). *If for some $l \in \mathbb{N}_0$, for any $\mathcal{I} \subset \mathbb{N}_l := \{0, 1, \dots, l\}$ and any pairwise distinct nodes $x_1, \dots, x_{n_{\mathcal{I}}} \in \mathbb{R}$, there exists a polynomial $p \in \mathcal{P}_{\mathcal{I}}^1$ such that*

$$p(x_j) = 0 \quad \text{for } j = 1, \dots, n_{\mathcal{I}} - 1, \text{ and} \quad (\text{B.25})$$

$$p(x_{n_{\mathcal{I}}}) = 1, \quad (\text{B.26})$$

then there also exists for any $\mathcal{I} \subset \mathbb{N}_l$ a unique polynomial $\bar{p} \in \mathcal{P}_{\mathcal{I}}^1$ such that

$$\bar{p}(x_j) = f_j, \quad \text{for } j = 1, \dots, n_{\mathcal{I}}, \quad (\text{B.27})$$

for arbitrary interpolation values $f_j \in \mathbb{R}$.

Proof. Let $i_1 < i_2 < \dots < i_{n_{\mathcal{I}}}$ be the elements of a given $\mathcal{I} \subset \mathbb{N}_l$. For $k = 1, \dots, n_{\mathcal{I}}$, we define the set $\mathcal{I}_k$ of the smallest k of these elements. For each of these sets $\mathcal{I}_k$, let p_k be the polynomial in $\mathcal{P}_{\mathcal{I}_k}^1$ that satisfies

$$p_k(x_j) = 0 \quad \text{for } j = 1, \dots, k-1, \text{ and} \quad (\text{B.28})$$

$$p_k(x_k) = 1. \quad (\text{B.29})$$

To find the polynomial $\bar{p} \in \mathcal{P}_{\mathcal{I}}^1$ that satisfies (B.27), we make the Ansatz

$$\bar{p}(x) = \sum_{k=1}^{n_{\mathcal{I}}} c_k p_k, \quad (\text{B.30})$$

to obtain an interpolation problem described by the linear system

$$\mathbf{A}\mathbf{c} = \mathbf{f}, \quad (\text{B.31})$$

with $\mathbf{c} = [c_1 \ \dots \ c_{n_{\mathcal{I}}}]^T \in \mathbb{R}^{n_{\mathcal{I}}}$, $\mathbf{f} = [f_1 \ \dots \ f_{n_{\mathcal{I}}}]^T \in \mathbb{R}^{n_{\mathcal{I}}}$ and the entries of the matrix $\mathbf{A} \in \mathbb{R}^{n_{\mathcal{I}} \times n_{\mathcal{I}}}$,

$$\mathbf{A}_{jk} := p_k(x_j), \quad j, k = 1, \dots, n_{\mathcal{I}}. \quad (\text{B.32})$$

Because of (B.28) and (B.29), $\mathbf{A}$ is a lower triangular matrix with entries 1 on its diagonal. Hence, $\mathbf{A}$ has full rank and the linear system (B.31) has a unique solution $\mathbf{c}$ for any right-hand side $\mathbf{f}$. $\square$

The following two lemmas extend this result to arbitrary degrees l ; Lemma B.6 for the case that all nodes x_j are positive, Lemma B.7 for the case that $x_{n_{\mathcal{I}}} = 0$ and $0 \in \mathcal{I}$.

Lemma B.6 (Specific gappy univariate interpolation for arbitrary l). *For any $l \in \mathbb{N}_0$, for any $\mathcal{I} \in \mathbb{N}_l := \{0, 1, \dots, l\}$, there exists a polynomial $p \in \mathcal{P}_{\mathcal{I}}^1$ such that the interpolation properties (B.25) and (B.26) hold for any pairwise distinct, positive nodes $x_1, \dots, x_{n_{\mathcal{I}}} \in \mathbb{R}^+$.*

Proof. The proof proceeds by induction along l .

- Base case: For $l = 0$, $p(x) = 1$ satisfies the interpolation properties for the only existing set $\mathcal{I} = \{0\}$.
- Induction hypothesis: For fixed l , there exists $p \in \mathcal{P}_{\mathcal{I}}^1$ such that the interpolation properties (B.25) and (B.26) hold for any $\mathcal{I} \in \mathbb{N}_l$ and any pairwise distinct, positive nodes $x_1, \dots, x_{n_{\mathcal{I}}} \in \mathbb{R}$.
- Induction step: We consider a set $\mathcal{I} \subset \mathbb{N}_{l+1} := \{0, 1, \dots, l, l+1\}$ and the pairwise distinct, positive nodes $x_1, \dots, x_{n_{\mathcal{I}}} \in \mathbb{R}^+$. If $l+1 \notin \mathcal{I}$, then the induction hypothesis directly applies. Otherwise, if $l+1 \in \mathcal{I}$, then the induction hypothesis holds for $\mathcal{I}^- := \mathcal{I} \setminus \{l+1\}$ and the nodes $x_1, \dots, x_{n_{\mathcal{I}}-1}$. Hence, Lemma B.5 implies that there exists a polynomial $\bar{p} \in \mathcal{P}_{\mathcal{I}^-}^1$ such that

$$\bar{p}(x_j) = f_j, \quad \text{for } j = 1, \dots, n_{\mathcal{I}} - 1, \quad (\text{B.33})$$

for arbitrary interpolation values $f_i \in \mathbb{R}$. Consequently, there exists $\bar{p} \in \mathcal{P}_{\mathcal{I}^-}^1$ such that

$$\bar{p}(x_j) = -x_j^{l+1}, \quad \text{for } j = 1, \dots, n_{\mathcal{I}} - 1. \quad (\text{B.34})$$

Then,

$$\hat{p}(x) := \bar{p}(x) + x^{l+1} \quad (\text{B.35})$$

satisfies (B.25), i.e., $x_1, \dots, x_{n_{\mathcal{I}}-1}$ are roots of $\hat{p}$. By Descartes' rule of signs, $\hat{p}$ has at most $n_{\mathcal{I}} - 1$ positive roots, hence $\hat{p}(x_{n_{\mathcal{I}}}) \neq 0$. Consequently, we can normalize $\hat{p}$ to find the desired polynomial,

$$p(x) := \frac{\hat{p}(x)}{\hat{p}(x_{n_{\mathcal{I}}})}. \quad (\text{B.36})$$

□

Lemma B.7 (Specific gappy univariate interpolation for arbitrary l with $x_{n_{\mathcal{I}}} = 0$). *For any $l \in \mathbb{N}_0 := \{0, 1, \dots, l\}$, for any $\mathcal{I} \in \mathbb{N}_l$ with $0 \in \mathcal{I}$, there exists a polynomial $p \in \mathcal{P}_{\mathcal{I}}^1$ such that the interpolation properties (B.25) and (B.26) hold for any pairwise distinct, positive nodes $x_1, \dots, x_{n_{\mathcal{I}}-1} \in \mathbb{R}^+$ and $x_{n_{\mathcal{I}}} = 0$.*

Proof. Let $\mathcal{I}^- := \mathcal{I} \setminus \{0\}$. According to Lemma B.5 and Lemma B.6, there exists a polynomial $\bar{p} \in \mathcal{P}_{\mathcal{I}^-}^1$ such that

$$\bar{p}(x_j) = -1 \quad \text{for } j = 1, \dots, n_{\mathcal{I}} - 1. \quad (\text{B.37})$$

Then,

$$p(x) := \bar{p}(x) + 1 \quad (\text{B.38})$$

is element of $\mathcal{P}_{\mathcal{I}}^1$ and satisfies (B.25). Because $\bar{p} \in \mathcal{P}_{\mathcal{I}^-}^1$ with $0 \notin \mathcal{I}^-$, x is a factor of $\bar{p}$, so $\bar{p}(0) = 0$. Consequently p also satisfies (B.26). $\square$

With these preparations, we can prove Lemma B.4.

Proof of Lemma B.4. For any $l \in \mathbb{N}$, for any $\hat{\mathcal{I}} = \{i^*\} \cup \mathcal{I}^- \subset \mathbb{N}_l$ with $i^* < \min \mathcal{I}^-$, let $\mathcal{I} = \{i - i^* \mid i \in \hat{\mathcal{I}}\}$ and $\{x_1, \dots, x_{n_{\mathcal{I}}}\} = \hat{\mathcal{I}}$. Then Lemma B.6 implies the statement if $i^* \neq 0$ and Lemma B.7 if $i^* = 0$. $\square$

B.1.5 Gappy multivariate polynomial interpolation

Now we have all ingredients to prove Theorem B.1.

Proof of Theorem B.1. The proof is performed by induction along the number of elements in $\mathcal{I}$.

- Base case: Let $\mathcal{I} = \{i\}$ for any $i \in \mathbb{N}_l$ with arbitrary $l \in \mathbb{N}_0$, then Theorem B.3 implies the existence and uniqueness of a polynomial $p \in \mathcal{P}_{\{i\}}^n$ that satisfies the interpolation property (B.5) in the interpolation nodes in $\mathcal{X}^i$.
- Induction hypothesis: Let for a fixed degree set $\mathcal{I}^- \subset \mathbb{N}_l$ with arbitrary $l \in \mathbb{N}_0$ exist a unique polynomial $p^- \in \mathcal{P}_{\mathcal{I}^-}^n$ that satisfies the interpolation property (B.5) in the interpolation nodes $\cup_{i \in \mathcal{I}^-} \mathcal{X}^i$.
- Induction step: We consider the degree set $\mathcal{I} := \{i^*\} \cup \mathcal{I}^-$ with arbitrary $i^* \in \mathbb{N}_l$. Without loss of generality, we can assume $i^* < \min \mathcal{I}^-$. As described in Section B.1.4, we can construct a polynomial

$$p(q) = p_{i^*}(q) \hat{p} \left(\sum_{k=1}^n q_k \right) \quad \forall q \in \mathbb{R}^n, \quad (\text{B.39})$$

with $p_{i^*} \in \mathcal{P}_{i^*}^n$ and $\hat{p} \in \mathcal{P}_{\mathcal{I}^-}^1$, so $p \in \mathcal{P}_{\mathcal{I}}^n$. Because of Lemma B.4 and $\sum_{k=1}^n q_k = i$ for all $\mathbf{q} \in \mathcal{X}^i$ for any $i \in \mathbb{N}_0$, we can choose $\hat{p}$ such that

$$\hat{p} \left(\sum_{k=1}^n q_k \right) = 0, \quad \text{for all } \mathbf{q} \in \mathcal{X}^i \text{ for all } i \in \mathcal{I}^- \quad (\text{B.40})$$

$$\hat{p} \left(\sum_{k=1}^n q_k \right) = 1, \quad \text{for all } \mathbf{q} \in \mathcal{X}^{i^*}, \quad (\text{B.41})$$

and p_{i^*} such that

$$p_{i^*}(\mathbf{q}_j) = b_j - p^-(\mathbf{q}_j), \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^{i^*}, \quad (\text{B.42})$$

with the interpolation values b_j prescribed in (B.5). Consequently, the polynomial p satisfies

$$p(\mathbf{q}_j) = 0, \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^i \text{ for all } i \in \mathcal{I}^- \text{ and} \quad (\text{B.43})$$

$$p(\mathbf{q}_j) = b_j - p^-(\mathbf{q}_j), \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^{i^*}. \quad (\text{B.44})$$

As a result, the polynomial $p^- + p$ satisfies

$$p^-(\mathbf{q}_j) + p(\mathbf{q}_j) = b_j + 0 = b_j, \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^i \text{ for all } i \in \mathcal{I}^- \text{ and} \quad (\text{B.45})$$

$$p^-(\mathbf{q}_j) = p(\mathbf{q}_j) = p^-(\mathbf{q}_j) + b_j - p^-(\mathbf{q}_j) = b_j, \quad \text{for all } \mathbf{q}_j \in \mathcal{X}^{i^*}, \quad (\text{B.46})$$

and hence satisfies the interpolation property (B.5). Since the corresponding interpolation matrix is square, existence of a solution also implies uniqueness. □

B.2 ALGORITHM FOR GENERATING ROM STATE SET

We present two different approaches how to implement the procedure to generate the rank-ensuring ROM state sets $\mathcal{X}^i$ as defined in (3.14) in line 7 of Algorithm 3.1. As shown in Algorithm B.1 for polynomial degree $i = 3$, the procedure essentially comprises 3 nested **for**-loops. The first approach shown in Algorithm B.2 constructs these nested **for**-loops for arbitrary i via recursion. This recursion makes the algorithm rather non-intuitive. While this non-intuitivity might be unavoidable to handle arbitrary polynomial degrees i , a more intuitive approach is possible in practice. In practice, we know a priori which polynomial degrees i are involved in the considered problems. For example, in the numerical

experiments in Section 3.4, the only polynomial degrees involved are $\{1, 2, 3, 8\}$. Consequently, we can use the hard-coded discrimination of cases shown in Algorithm B.3 as alternative to Algorithm B.2, with the procedures for $i = 1, 2, 8$ implemented analogous to Algorithm B.1.

Algorithm B.1 Generate rank-ensuring ROM state set $\mathcal{X}^3$

```

1: procedure GENERATEROMSTATESET3( ROM dimension  $n$  )
2:   Initialize  $\mathcal{X}^3 \leftarrow \emptyset$ 
3:   for  $j_1 = 1, \dots, n$  do
4:     for  $j_2 = j_1, \dots, n$  do
5:       for  $j_3 = j_2, \dots, n$  do
6:          $\mathcal{X}^3 \leftarrow \mathcal{X}^3 \cup e_{j_1} + e_{j_2} + e_{j_3}$ 
7:       end for
8:     end for
9:   end for
10:  return  $\mathcal{X}^3$ 
11: end procedure

```

B.3 ROM STATE ERRORS WITH RESPECT TO THE FOM

Figures B.1, B.2 and B.3 show the relative average ROM state errors,

$$\frac{\|\mathbf{X} - \mathbf{V}_n \tilde{\mathbf{X}}\|_F}{\|\mathbf{X}\|_F}, \quad (\text{B.47})$$

for Chafee-Infante equation in Section 3.4.1, ice sheet model in Section 3.4.2 and Burgers' equation in Section 3.4.3, respectively. The matrix $\mathbf{X} = [\mathbf{x}_0 \ \dots \ \mathbf{x}_{K_{\text{POD}}-1}]$ comprises the FOM snapshots used to construct the POD basis, and $\tilde{\mathbf{X}} = [\tilde{\mathbf{x}}_0 \ \dots \ \tilde{\mathbf{x}}_{K_{\text{POD}}-1}]$ the corresponding ROM snapshots generated by integrating the ROM ODE (3.6) with the intrusive ROM operators and the inferred ROM operators, respectively. As can be seen, the ROM errors decrease monotonously for increasing ROM dimensions in all three test cases for both, intrusive and inferred ROMs. Additionally, we cannot see any difference between the errors of corresponding intrusive and inferred ROMs. In fact, the differences between these errors are all smaller than 10^{-10} . This alignment is due to the exact reconstruction of the intrusive ROM operators shown in Fig. 3.1.

Algorithm B.2 Generate rank-ensuring ROM state set $\mathcal{X}^i$ - recursive algorithm

```

1: procedure GENERATEROMSTATESET(
    ROM dimension  $n$ ,
    polynomial degree  $i \in \mathbb{N}_0$ 
)
2:   return RECURSIVEROMSTATESET( ROM dimension  $n$ ,
                                   polynomial degree  $i$ ,
                                   current for loop depth  $i^* = 0$ ,
                                   current index sequence  $\mathbf{j} = []$ ,
                                   current ROM state set  $\mathcal{X}^i = \emptyset$  )
3: end procedure

4: procedure RECURSIVEROMSTATESET( ROM dimension  $n$ ,
    polynomial degree  $i \in \mathbb{N}_0$ ,
    current for loop depth  $i^*$ ,
    current index sequence  $\mathbf{j} = [j_1, \dots, j_{i^*}]$ ,
    # always has length  $i^*$ 
    current ROM state set  $\mathcal{X}^i$  )
5:   if  $i^* = i$  then
6:     Initialize  $\hat{\mathbf{x}} \leftarrow \mathbf{0} \in \mathbb{R}^n$ 
7:     for  $k=1, \dots, i$  do
8:        $\hat{\mathbf{x}} \leftarrow \hat{\mathbf{x}} + \mathbf{e}_{j_k}$ 
9:     end for
10:    return  $\mathcal{X}^i \cup \hat{\mathbf{x}}$ 
11:   else
12:     # Perform next inner for-loop
13:     for  $j_{i^*+1} = j_{i^*}, \dots, n$  do # with  $j_0 = 1$ 
14:       return RECURSIVEROMSTATESET(
        ROM dimension  $n$ ,
        polynomial degree  $i$ ,
        current for loop depth  $i^* + 1$ ,
        current index sequence  $[j_1, \dots, j_{i^*+1}]$ ,
        current ROM state set  $\mathcal{X}^i$  )
15:     end for
16:   end if
17: end procedure

```

Algorithm B.3 Generate rank-ensuring ROM state set $\mathcal{X}^i$ - hard-coded

```

1: procedure GENERATEROMSTATESET(
    ROM dimension  $n$ ,
    polynomial degree  $i \in \mathbb{N}_0$ 
)
2:   if  $i = 1$  then return GENERATEROMSTATESET1(
        ROM dimension  $n$ )
3:   else if  $i = 2$  then return GENERATEROMSTATESET2(
        ROM dimension  $n$ )
4:   else if  $i = 3$  then return GENERATEROMSTATESET3(
        ROM dimension  $n$ )
5:   else if  $i = 8$  then return GENERATEROMSTATESET8(
        ROM dimension  $n$ )
6:   end if
7: end procedure

```

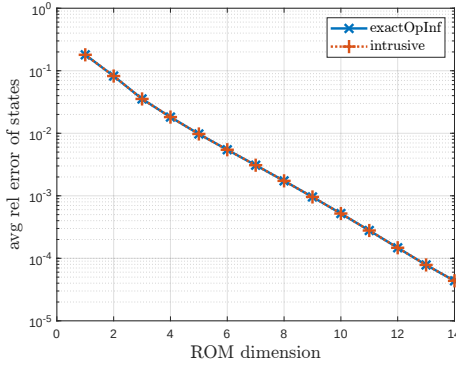


Figure B.1: Relative average ROM state error (B.47) for Chafee-Infante equation.

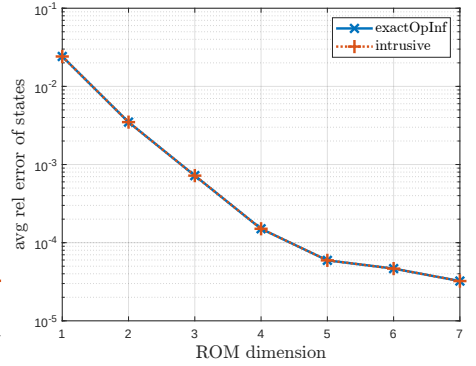


Figure B.2: Relative average ROM state error (B.47) for the ice sheet model.

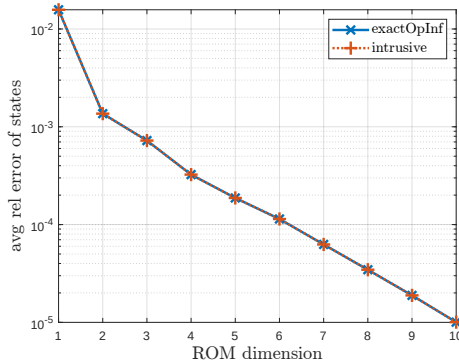


Figure B.3: Relative average ROM state error (B.47) for Burgers' equation.

SUPPLEMENTARY MATERIAL FOR CHAPTER 4

C.1 DECOUPLING FOR ARBITRARY POLYNOMIAL RIGHT-HAND SIDE

The decoupling described in Section 4.4 can be performed analogously, if the right-hand side of (4.1) is any monomial in x other than quadratic. If this right-hand side consists of more than one monomial, the decoupling can be performed as follows.

C.1.1 Parametric dynamical system and Galerkin ROM

Let us assume that the right-hand side consists of a constant, a linear and a quadratic term,

$$f(x; \mu) = c(\mu) + \mathbf{A}(\mu)x + \mathbf{H}(\mu)x^2, \quad (\text{C.1})$$

with operators $c(\mu) \in \mathbb{R}^N$, $\mathbf{A}(\mu) \in \mathbb{R}^{N \times N}$ and $\mathbf{H}(\mu) \in \mathbb{R}^{N \times \binom{N+1}{2}}$ with affine decompositions,

$$\begin{aligned} c(\mu) &= \sum_{p=1}^{q_c} \theta_c^{(p)}(\mu) c^{(p)}, \quad \mathbf{A}(\mu) = \sum_{p=1}^{q_A} \theta_A^{(p)}(\mu) \mathbf{A}^{(p)}, \\ \mathbf{H}(\mu) &= \sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \mathbf{H}^{(p)}, \end{aligned} \quad (\text{C.2})$$

with scalar functions $\theta_c^{(p)}$, $\theta_A^{(p)}$ and $\theta_H^{(p)}$ and matrices $c^{(p)}$, $\mathbf{A}^{(p)}$ and $\mathbf{H}^{(p)}$ of the same dimensions as $c(\mu)$, $\mathbf{A}(\mu)$ and $\mathbf{H}(\mu)$, respectively. The intrusive ROM operators are defined as

$$\begin{aligned} \tilde{c}(\mu) &= \mathbf{V}_n^T c(\mu) \in \mathbb{R}^{n \times 1}, \quad \tilde{\mathbf{A}}(\mu) = \mathbf{V}_n^T \mathbf{A}(\mu) \mathbf{V}_n \in \mathbb{R}^{n \times n} \\ \text{and } \tilde{\mathbf{H}}(\mu) &= \mathbf{V}_n^T \mathbf{H}(\mu) \mathbf{I}_N^{(i)} (\mathbf{V}_n \otimes \mathbf{V}_n) \mathbf{J}_n^{(i)}. \end{aligned} \quad (\text{C.3})$$

C.1.2 Parametric operator inference

As described in [70, 119], the operator inference problem to infer the operator matrix

$$\hat{\mathbf{O}} := \left[\hat{c}^{(1)} \dots \hat{c}^{(q_c)} \mid \hat{\mathbf{A}}^{(1)} \dots \hat{\mathbf{A}}^{(q_A)} \mid \hat{\mathbf{H}}^{(1)} \dots \hat{\mathbf{H}}^{(q_H)} \right] \in \mathbb{R}^{n \times q(n)}, \quad (\text{C.4})$$

with $q(n) = q_c + q_A n + q_H \binom{n+1}{2}$ is then to minimize $\mathcal{L}(\hat{\mathbf{O}})$ defined in (4.14), but with

$$\mathbf{D} = [\quad \mathbf{D}_c \quad | \quad \mathbf{D}_A \quad | \quad \mathbf{D}_H \quad] \quad (\text{C.5})$$

$$= \left[\begin{array}{c|c|c} \boldsymbol{\theta}_c(\boldsymbol{\mu}_1) \otimes \mathbf{1}_K & \boldsymbol{\theta}_A(\boldsymbol{\mu}_1) \otimes \hat{\mathbf{X}}(\boldsymbol{\mu}_1)^T & \boldsymbol{\theta}_H(\boldsymbol{\mu}_1) \otimes \hat{\mathbf{X}}_2(\boldsymbol{\mu}_1)^T \\ \vdots & \vdots & \vdots \\ \boldsymbol{\theta}_c(\boldsymbol{\mu}_s) \otimes \mathbf{1}_K & \boldsymbol{\theta}_A(\boldsymbol{\mu}_s) \otimes \hat{\mathbf{X}}(\boldsymbol{\mu}_s)^T & \boldsymbol{\theta}_H(\boldsymbol{\mu}_s) \otimes \hat{\mathbf{X}}_2(\boldsymbol{\mu}_s)^T \end{array} \right], \quad (\text{C.6})$$

and the row vectors that encode the affine parameter dependencies in (C.2),

$$\boldsymbol{\theta}_c(\boldsymbol{\mu}_i) = [\theta_c^{(1)}(\boldsymbol{\mu}_i) \quad \dots \quad \theta_c^{(q_c)}(\boldsymbol{\mu}_i)] \in \mathbb{R}^{1 \times q_c} \quad (\text{C.7})$$

$$\boldsymbol{\theta}_A(\boldsymbol{\mu}_i) = [\theta_A^{(1)}(\boldsymbol{\mu}_i) \quad \dots \quad \theta_A^{(q_A)}(\boldsymbol{\mu}_i)] \in \mathbb{R}^{1 \times q_A} \quad (\text{C.8})$$

$$\boldsymbol{\theta}_H(\boldsymbol{\mu}_i) = [\theta_H^{(1)}(\boldsymbol{\mu}_i) \quad \dots \quad \theta_H^{(q_H)}(\boldsymbol{\mu}_i)] \in \mathbb{R}^{1 \times q_H}. \quad (\text{C.9})$$

C.1.3 Decoupling parametric operator inference

The parametric operator inference minimization problem (4.14) is high-dimensional because it involves data from s parameter samples to learn q_c , q_A and q_H constant, linear and quadratic operators, respectively. We want to decouple this problem into subproblems that only involve data from one parameter sample to learn a set of one constant, one linear and one quadratic operator. For this purpose, we first note that

$$\mathcal{L}(\hat{\mathbf{O}}) = \sum_{i=1}^s \left\| \mathbf{D}^{(i)}(\boldsymbol{\mu}_i) \left[\begin{array}{c} \sum_{p=1}^{q_c} \theta_c^{(p)}(\boldsymbol{\mu}_i) \hat{\mathbf{c}}^{(p)} \\ \sum_{p=1}^{q_A} \theta_A^{(p)}(\boldsymbol{\mu}_i) \hat{\mathbf{A}}^{(p)} \\ \sum_{p=1}^{q_H} \theta_H^{(p)}(\boldsymbol{\mu}_i) \hat{\mathbf{H}}^{(p)} \end{array} \right]^T - \hat{\mathbf{X}}(\boldsymbol{\mu}_i) \right\|_F^2, \quad (\text{C.10})$$

with

$$\mathbf{D}^{(i)}(\boldsymbol{\mu}_i) := [\mathbf{1}_K \mid \hat{\mathbf{X}}(\boldsymbol{\mu}_i) \mid \hat{\mathbf{X}}_2(\boldsymbol{\mu}_i)] \in \mathbb{R}^{K \times q(n)} \quad (\text{C.11})$$

$$\text{and } \mathbf{1}_K = [1 \quad \dots \quad 1]^T \in \mathbb{R}^K. \quad (\text{C.12})$$

If $\boldsymbol{\Theta}_c = \boldsymbol{\Theta}_A = \boldsymbol{\Theta}_H = \mathbf{I}_s$ (so $q_c = q_A = q_H = s$), then the term on the right simplifies to

$$\sum_{i=1}^s \left\| \mathbf{D}^{(i)}(\boldsymbol{\mu}_i) \left[\begin{array}{c} \theta_c^{(i)}(\boldsymbol{\mu}_i) \hat{\mathbf{c}}^{(i)} \\ \theta_A^{(i)}(\boldsymbol{\mu}_i) \hat{\mathbf{A}}^{(i)} \\ \theta_H^{(i)}(\boldsymbol{\mu}_i) \hat{\mathbf{H}}^{(i)} \end{array} \right]^T - \hat{\mathbf{X}}(\boldsymbol{\mu}_i) \right\|_F^2$$

so for all $i = 1, \dots, s$, the operators $\hat{\mathbf{c}}^{(i)}, \hat{\mathbf{A}}^{(i)}, \hat{\mathbf{H}}^{(i)}$ only depend on data from the parameter sample $\boldsymbol{\mu}_i$. Hence, we can solve subproblems for the basis operators $\hat{\mathbf{O}}^{(i)} = [\hat{\mathbf{c}}^{(i)} | \hat{\mathbf{A}}^{(i)} | \hat{\mathbf{H}}^{(i)}]$ separately.

In general, $\boldsymbol{\Theta}_c = \boldsymbol{\Theta}_A = \boldsymbol{\Theta}_H = \mathbf{I}_s$ does not hold (i.e., we cannot find suitable parameter samples $\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_s$). However, we can achieve similar decoupling if the rank-ensuring snapshot data described in Section 4.3 are used and the matrices $\boldsymbol{\Theta}_c, \boldsymbol{\Theta}_A$ and $\boldsymbol{\Theta}_H$ have full rank. For this purpose, we first note that the matrices $\boldsymbol{\Theta}_c \in \mathbb{R}^{s \times q_c}$, $\boldsymbol{\Theta}_A \in \mathbb{R}^{s \times q_A}$ and $\boldsymbol{\Theta}_H \in \mathbb{R}^{s \times q_H}$ can be extended to square full rank matrices of dimension $s \times s$. Such an extension corresponds to extending the affine expansions (4.5) by additional terms that evaluate to zero. Correspondingly, we extend the operator matrix $\hat{\mathbf{O}}$ to

$$\hat{\mathbf{O}} = \left[\hat{\mathbf{c}}^{(1)} \dots \hat{\mathbf{c}}^{(s)} \mid \hat{\mathbf{A}}^{(1)} \dots \hat{\mathbf{A}}^{(s)} \mid \hat{\mathbf{H}}^{(1)} \dots \hat{\mathbf{H}}^{(s)} \right] \quad (\text{C.13})$$

$$\in \mathbb{R}^{n \times (1+n+\binom{n+1}{2})s} \quad (\text{C.14})$$

with $\hat{\mathbf{c}}^{(i)} = \mathbf{0}$ for all $i > q_c$, $\hat{\mathbf{A}}^{(i)} = \mathbf{0}$ for all $i > q_A$ and $\hat{\mathbf{H}}^{(i)} = \mathbf{0}$ for all $i > q_H$.

Next we note that the snapshot data $\hat{\mathbf{X}}(\boldsymbol{\mu}_i) = \hat{\mathbf{X}}$ described in Section 4.3 is parameter-independent, so the data matrix $\mathbf{D}$ simplifies to

$$\mathbf{D} = \left[\boldsymbol{\Theta}_c \otimes \mathbf{1}_K \mid \boldsymbol{\Theta}_A \otimes \hat{\mathbf{X}}^T \mid \boldsymbol{\Theta}_H \otimes \hat{\mathbf{X}}_2^T \right]. \quad (\text{C.15})$$

Now we can define

$$\check{\mathbf{O}} = \hat{\mathbf{O}} \mathbf{T}^{-T}, \quad \text{with} \quad \mathbf{T} = \begin{bmatrix} \boldsymbol{\Theta}_c^{-1} \otimes \mathbf{I}_1 & & \\ & \boldsymbol{\Theta}_A^{-1} \otimes \mathbf{I}_n & \\ & & \boldsymbol{\Theta}_H^{-1} \otimes \mathbf{I}_{\binom{n+1}{2}} \end{bmatrix}, \quad (\text{C.16})$$

then

$$\mathbf{D}\widehat{\mathbf{O}}^T = \mathbf{D} \left(\check{\mathbf{O}}^T \right)^T = \mathbf{D}\mathbf{T}\check{\mathbf{O}}^T \quad (\text{C.17})$$

$$= \left[\begin{array}{c|c|c} \boldsymbol{\Theta}_c \otimes \mathbf{1}_K & \boldsymbol{\Theta}_A \otimes \widehat{\mathbf{X}}^T & \boldsymbol{\Theta}_H \otimes \widehat{\mathbf{X}}_2^T \end{array} \right] \left[\begin{array}{c} \boldsymbol{\Theta}_c^{-1} \otimes \mathbf{I}_1 \\ \boldsymbol{\Theta}_A^{-1} \otimes \mathbf{I}_n \\ \boldsymbol{\Theta}_H^{-1} \otimes \mathbf{I}_{\binom{n+1}{2}} \end{array} \right] \check{\mathbf{O}}^T \quad (\text{C.18})$$

$$= \left[\begin{array}{c|c} (\boldsymbol{\Theta}_c \boldsymbol{\Theta}_c^{-1}) \otimes (\mathbf{1}_K \mathbf{I}_1) & (\boldsymbol{\Theta}_A \boldsymbol{\Theta}_A^{-1}) \otimes (\widehat{\mathbf{X}}^T \mathbf{I}_n) \\ & (\boldsymbol{\Theta}_H \boldsymbol{\Theta}_H^{-1}) \otimes (\widehat{\mathbf{X}}_2^T \mathbf{I}_{\binom{n+1}{2}}) \end{array} \right] \check{\mathbf{O}}^T \quad (\text{C.19})$$

$$= \left[\begin{array}{c|c|c} \mathbf{I}_s \otimes \mathbf{1}_K & \mathbf{I}_s \otimes \widehat{\mathbf{X}}^T & \mathbf{I}_s \otimes \widehat{\mathbf{X}}_2^T \end{array} \right] \check{\mathbf{O}}^T \quad (\text{C.20})$$

Hence, we can decouple the problem

$$\mathcal{L}(\check{\mathbf{O}}) = \left\| \check{\mathbf{D}}\check{\mathbf{O}}^T - \mathbf{R}^T \right\|_F^2 \quad (\text{C.21})$$

with $\check{\mathbf{D}} = \mathbf{D}\mathbf{T}$.

This decoupling results in smaller, embarrassingly parallel and numerically more stable subproblems

$$\mathcal{L}(\check{\mathbf{O}}^{(i)}) = \left\| \check{\mathbf{D}}^{(i)} \left(\check{\mathbf{O}}^{(i)} \right)^T - \left(\mathbf{R}^{(i)}(\boldsymbol{\mu}_i) \right)^T \right\|_F^2, \quad i = 1, \dots, s \quad (\text{C.22})$$

$$\begin{aligned} \text{with } \check{\mathbf{D}}^{(i)} &= \left[\mathbf{1}_K | \widehat{\mathbf{X}}^T | \widehat{\mathbf{X}}_2^T \right], \quad \check{\mathbf{O}}^{(i)} = [\check{\mathbf{c}}^{(i)} | \check{\mathbf{A}}^{(i)} | \check{\mathbf{H}}^{(i)}] \\ \text{and } \mathbf{R}^{(i)}(\boldsymbol{\mu}_i) &= \hat{\mathbf{X}}(\boldsymbol{\mu}_i) \end{aligned} \quad (\text{C.23})$$

and facilitates extensions by new parameter samples/new parametric terms. The original operator $\widehat{\mathbf{O}}$ is then computed via $\widehat{\mathbf{O}} = \check{\mathbf{O}}\mathbf{T}^T$.

Note that the addition of zero terms in (C.14) is not necessary for exact operator inference. For example, if $q_c > q_A > q_H$, it would be sufficient to compute exact snapshot data for $\mathcal{I} = \{0, 1, 2\}$ for q_H parameter samples, for $\mathcal{I} = \{0, 1\}$ for $q_A - q_H$ samples and for $\mathcal{I} = \{0\}$ for $q_c - q_A$ samples — provided that the corresponding matrix $\mathbf{D}$ has full rank. While this approach would reduce the demand for generation of snapshot data, it would destroy the parameter independency $\widehat{\mathbf{X}}(\boldsymbol{\mu}_i) = \widehat{\mathbf{X}}$ and hence make the decoupling impossible.

C.2 ROM STATE ERRORS WITH RESPECT TO THE FOM

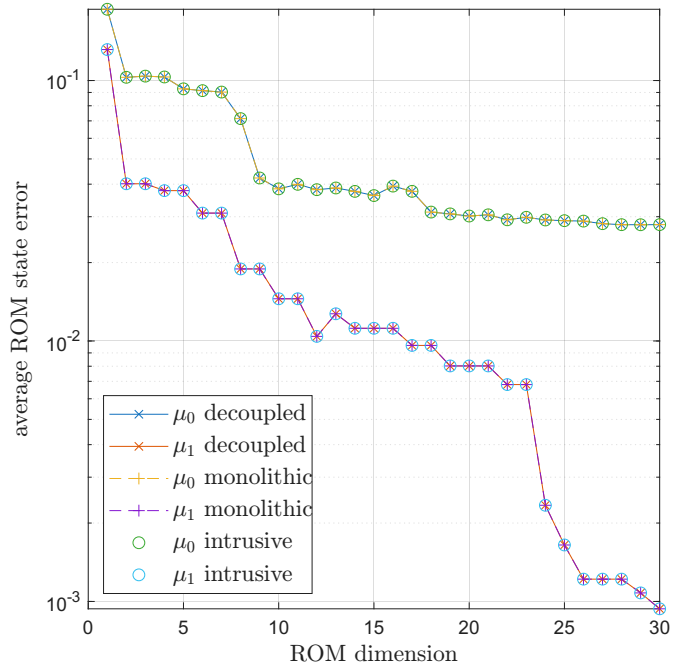
Figs. C.1a and C.1b show the average ROM state error with respect to the FOM,

$$\frac{\sum_{k=0}^K \|\mathbf{x}_k(\mu) - \mathbf{V}_n \tilde{\mathbf{x}}_k(\mu)\|_2}{\sum_{k=0}^K \|\mathbf{x}_k(\mu)\|_2}, \quad (\text{C.24})$$

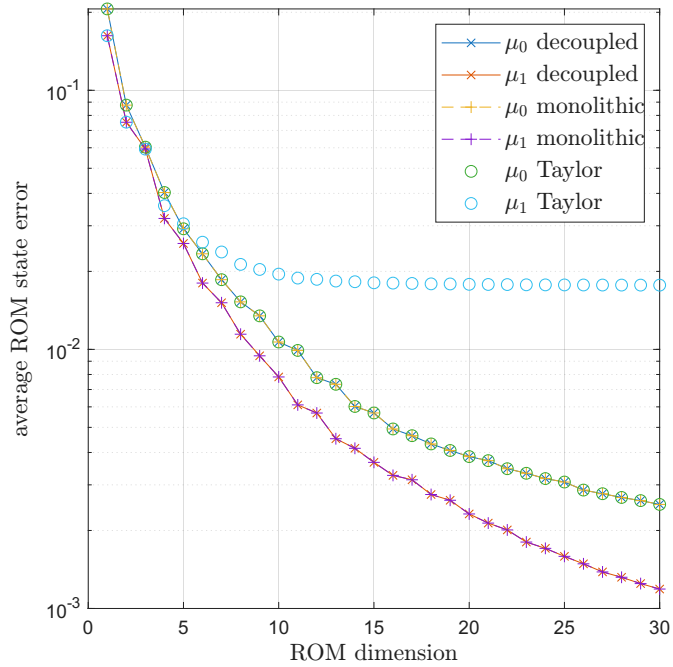
where $\tilde{\mathbf{x}}_k$ are the snapshots of the explicit Euler ROM

$$\begin{aligned} \tilde{\mathbf{x}}_{k+1}(\mu) &= \tilde{\mathbf{x}}_k(\mu) + \Delta t \left(\sum_{p=1}^{q_H} \theta_H^{(p)}(\mu) \bar{\mathbf{H}}^{(p)} \right) \tilde{\mathbf{x}}_k^2(\mu), \\ k &= 0, \dots, K-1, \end{aligned} \quad (\text{C.25})$$

with the ROM operators $\bar{\mathbf{H}}^{(p)}$ computed intrusively, or via monolithic or decoupled operator inference, respectively. As can be seen in Fig. C.1a for the affine test case, decoupled, monolithic and intrusive ROM align for both parameter values μ_0 and μ_1 . The errors decrease for increasing ROM dimension, but not strictly monotonically and for μ_0 much slower than for μ_1 . In Fig. C.1b for the non-affine test case, decoupled and monolithic ROM align as well, but the Taylor-based intrusive ROM only for μ_0 . This mismatch is in line with the observed mismatch of operator errors in Fig. 4.2a. Also for this test case, all errors decrease with increasing ROM dimension, but for the Taylor-based intrusive ROM this decrease seems stalled for μ_1 to magnitude 10^{-2} .



(a) Test case with affine conductivity (4.41).



(b) Test case with non-affine conductivity (4.43).

Figure C.1: Relative average ROM state error.

BIBLIOGRAPHY

- [1] Babak Maboudi Afkham and Jan S Hesthaven. "Structure preserving model reduction of parametric Hamiltonian systems." In: *SIAM Journal on Scientific Computing* 39.6 (2017), A2616–A2644.
- [2] Syver Døving Agdestein and Benjamin Sanderse. "Discretize first, filter next: Learning divergence-consistent closure models for large-eddy simulation." In: *Journal of Computational Physics* 522 (2025), p. 113577.
- [3] Shady E Ahmed, Suraj Pawar, Omer San, Adil Rasheed, Traian Iliescu, and Bernd R Noack. "On closures for reduced order models—A spectrum of first-principle to machine-learned avenues." In: *Physics of Fluids* 33.9 (2021), p. 091301.
- [4] Robert Altmann and Jan Heiland. "Finite element decomposition and minimal extension for flow equations." In: *ESAIM: Mathematical Modelling and Numerical Analysis* 49.5 (2015), pp. 1489–1509.
- [5] Athanasios C Antoulas. *Approximation of large-scale dynamical systems*. SIAM, 2005.
- [6] Nicole Aretz and Karen Willcox. "Enforcing structure in data-driven reduced modeling through nested operator inference." In: *2024 IEEE 63rd Conference on Decision and Control (CDC)*. IEEE, 2024, pp. 8046–8053.
- [7] Maciej Balajewicz and Earl H Dowell. "Stabilization of projection-based reduced order models of the Navier–Stokes." In: *Nonlinear Dynamics* 70 (2012), pp. 1619–1632.
- [8] Maciej Balajewicz, Irina Tezaur, and Earl Dowell. "Minimal subspace rotation on the Stiefel manifold for stabilization and enhancement of projection-based reduced order models for the compressible Navier–Stokes equations." In: *Journal of Computational Physics* 321 (2016), pp. 224–241.
- [9] Francesco Ballarin, Andrea Manzoni, Alfio Quarteroni, and Gianluigi Rozza. "Supremizer stabilization of POD–Galerkin approximation of parametrized steady incompressible Navier–Stokes equations." In: *International Journal for Numerical Methods in Engineering* 102.5 (2015), pp. 1136–1161.

- [10] Matthew F Barone, Irina Kalashnikova, Daniel J Segalman, and Heidi K Thornquist. "Stable Galerkin reduced order models for linearized compressible flow." In: *Journal of Computational Physics* 228.6 (2009), pp. 1932–1946.
- [11] Peter Benner, Pawan Goyal, Jan Heiland, and Igor Pontes Duff. "Operator inference and physics-informed learning of low-dimensional models for incompressible flows." In: *Electronic Transactions on Numerical Analysis* 56 (2022), pp. 28–51. DOI: http://dx.doi.org/10.1553/etna_vol56s28.
- [12] Peter Benner, Serkan Gugercin, and Karen Willcox. "A survey of projection-based model reduction methods for parametric dynamical systems." In: *SIAM review* 57.4 (2015), pp. 483–531.
- [13] Peter Benner, Jens Saak, and M Monir Uddin. "Structure preserving model order reduction of large sparse second-order index-1 systems and application to a mechatronics model." In: *Mathematical and Computer Modelling of Dynamical Systems* 22.6 (2016), pp. 509–523.
- [14] Harsh Bhatia, Gregory Norgard, Valerio Pascucci, and Peer-Timo Bremer. "The Helmholtz-Hodge decomposition—a survey." In: *IEEE Transactions on visualization and computer graphics* 19.8 (2012), pp. 1386–1404.
- [15] Bart Blockmans, Tommaso Tamarozzi, Frank Naets, and Wim Desmet. "A nonlinear parametric model reduction method for efficient gear contact simulations." In: *International Journal for Numerical Methods in Engineering* 102.5 (2015), pp. 1162–1191.
- [16] Pascal den Boef, Diana Manvelyan-Stroot, Joseph Maubach, Wil Schilders, and Nathan van de Wouw. "Stable sparse operator inference for nonlinear structural dynamics." In: *Computers & Mathematics with Applications* 200 (2025), pp. 228–242.
- [17] Steven L Brunton and J Nathan Kutz. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.
- [18] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. "Discovering governing equations from data by sparse identification of nonlinear dynamical systems." In: *Proceedings of the national academy of sciences* 113.15 (2016), pp. 3932–3937.
- [19] Patrick Buchfink, Silke Glas, and Bernard Haasdonk. "Symplectic model reduction of Hamiltonian systems on nonlinear manifolds and approximation with weakly symplectic autoencoder." In: *SIAM Journal on Scientific Computing* 45.2 (2023), A289–A311.

- [20] Tan Bui-Thanh, Karen Willcox, and Omar Ghattas. "Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications." In: *AIAA journal* 46.10 (2008), pp. 2520–2529.
- [21] John Charles Butcher and Nicolette Goodwin. *Numerical methods for ordinary differential equations*. Vol. 2. Wiley Online Library, 2008.
- [22] Alfonso Caiazzo, Traian Iliescu, Volker John, and Svetlana Schyschlowa. "A numerical investigation of velocity–pressure reduced order models for incompressible flows." In: *Journal of Computational Physics* 259 (2014), pp. 598–616.
- [23] Kevin Carlberg, Matthew Barone, and Harbir Antil. "Galerkin v. least-squares Petrov–Galerkin projection in nonlinear model reduction." In: *Journal of Computational Physics* 330 (2017), pp. 693–734.
- [24] Kevin Carlberg, Charbel Farhat, Julien Cortial, and David Amallem. "The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows." In: *Journal of Computational Physics* 242 (2013), pp. 623–647.
- [25] Kevin Carlberg, Ray Tuminaro, and Paul Boggs. "Preserving Lagrangian structure in nonlinear model reduction with application to structural dynamics." In: *SIAM Journal on Scientific Computing* 37.2 (2015), B153–B184.
- [26] Jesse Chan. "Entropy stable reduced order modeling of nonlinear conservation laws." In: *Journal of Computational Physics* 423 (2020), p. 109789.
- [27] Krishan Chand, Henrik Rosenberger, and Benjamin Sanderse. "A pressure-free long-time stable reduced-order model for two-dimensional Rayleigh–Bénard convection." In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 34.2 (2024). DOI: [10.1063/5.0168857](https://doi.org/10.1063/5.0168857).
- [28] Saifon Chaturantabut, Chris Beattie, and Serkan Gugercin. "Structure-preserving model reduction for nonlinear port-Hamiltonian systems." In: *SIAM Journal on Scientific Computing* 38.5 (2016), B837–B865.
- [29] Saifon Chaturantabut and Danny C Sorensen. "Discrete empirical interpolation for nonlinear model reduction." In: *Proceedings of the 48th IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*. IEEE. 2009, pp. 4316–4321.

- [30] Joris Degroote, Jan Vierendeels, and Karen Willcox. "Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis." In: *International Journal for Numerical Methods in Fluids* 63.2 (2010), pp. 207–230.
- [31] Suchuan Dong. "A convective-like energy-stable open boundary condition for simulations of incompressible flows." In: *Journal of Computational Physics* 302 (2015), pp. 300–328.
- [32] Zlatko Drmac and Serkan Gugercin. "A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions." In: *SIAM Journal on Scientific Computing* 38.2 (2016), A631–A648.
- [33] Robert Eymard, Thierry Gallouët, and Raphaële Herbin. "Finite volume methods." In: *Solution of Equation in R_n (Part 3), Techniques of Scientific Computing (Part 3)*. Vol. 7. Handbook of Numerical Analysis. Elsevier, 2000, pp. 713–1018. DOI: [https://doi.org/10.1016/S1570-8659\(00\)07005-8](https://doi.org/10.1016/S1570-8659(00)07005-8). URL: <https://www.sciencedirect.com/science/article/pii/S1570865900070058>.
- [34] Ionut Farcas, Rayomand Gundevia, Ramakanth Munipalli, and Karen E Willcox. "Parametric non-intrusive reduced-order models via operator inference for large-scale rotating detonation engine simulations." In: *AIAA Scitech 2023 Forum*. 2023, p. 0172.
- [35] Charles L Fefferman. "Existence and smoothness of the Navier-Stokes equation." In: *The millennium prize problems* 57.67 (2006), p. 22.
- [36] Lambert Fick, Yvon Maday, Anthony T Patera, and Tommaso Taddei. "A reduced basis technique for long-time unsteady turbulent flows." In: *arXiv preprint arXiv:1710.03569* (2017).
- [37] Lambert Fick, Yvon Maday, Anthony T Patera, and Tommaso Taddei. "A stabilized POD model for turbulent flows over a range of Reynolds numbers: Optimal parameter sampling and constrained projection." In: *Journal of Computational Physics* 371 (2018), pp. 214–243.
- [38] Ciprian Foias, Oscar Manley, Ricardo Rosa, and Roger Temam. *Navier-Stokes equations and turbulence*. Vol. 83. Cambridge University Press, 2001.
- [39] Mariano Gasca. "Multivariate Polynomial Interpolation." In: *Computation of curves and surfaces*. Springer, 1990, pp. 215–236.
- [40] Mariano Gasca and Thomas Sauer. "Polynomial interpolation in several variables." In: *Advances in Computational Mathematics* 12 (2000), pp. 377–410.

- [41] Toby van Gastelen, Wouter Edeling, and Benjamin Sanderse. "Energy-conserving neural network for turbulence closure modeling." In: *Journal of Computational Physics* 508 (2024), p. 113003.
- [42] Anna-Lena Gerner and Karen Veroy. "Certified reduced basis methods for parametrized saddle point problems." In: *SIAM Journal on Scientific Computing* 34.5 (2012), A2812–A2836.
- [43] Michele Girfoglio, Annalisa Quaini, and Gianluigi Rozza. "A POD-Galerkin reduced order model for the Navier–Stokes equations in stream function-vorticity formulation." In: *Computers & Fluids* 244 (2022), p. 105536.
- [44] Yuezheng Gong, Qi Wang, and Zhu Wang. "Structure-preserving Galerkin POD reduced-order modeling of Hamiltonian systems." In: *Computer Methods in Applied Mechanics and Engineering* 315 (2017), pp. 780–798.
- [45] Pawan Goyal, Igor Pontes Duff, and Peter Benner. "Guaranteed stable quadratic models and their applications in SINDy and operator inference." In: *Physica D: Nonlinear Phenomena* (2025), p. 134893.
- [46] Carmen Gräßle, Michael Hinze, Jens Lang, and Sebastian Ullmann. "POD model order reduction with space-adapted snapshots for incompressible flows." In: *Advances in Computational Mathematics* 45.5 (2019), pp. 2401–2428.
- [47] Philip M Gresho and Robert L Sani. *Incompressible flow and the finite element method. Volume 1: Advection-diffusion and isothermal laminar flow*. John Wiley and Sons, 1998.
- [48] Anthony Gruber, Max Gunzburger, Lili Ju, and Zhu Wang. "Energetically consistent model reduction for metriplectic systems." In: *Computer Methods in Applied Mechanics and Engineering* 404 (2023), p. 115709.
- [49] Anthony Gruber and Irina Tezaur. "Canonical and noncanonical Hamiltonian operator inference." In: *Computer Methods in Applied Mechanics and Engineering* 416 (2023), p. 116334.
- [50] Max D Gunzburger, Janet S Peterson, and John N Shadid. "Reduced-order modeling of time-dependent PDEs with multiple parameters in the boundary data." In: *Computer methods in applied mechanics and engineering* 196.4-6 (2007), pp. 1030–1047.
- [51] Bernard Haasdonk and Mario Ohlberger. "Reduced basis method for finite volume approximations of parametrized linear evolution equations." In: *ESAIM: Mathematical Modelling and Numerical Analysis* 42.2 (2008), pp. 277–302.

- [52] Bernard Haasdonk and Mario Ohlberger. “Efficient reduced models and a posteriori error estimation for parametrized dynamical systems by offline/online decomposition.” In: *Mathematical and Computer Modelling of Dynamical Systems* 17.2 (2011), pp. 145–161.
- [53] Ernst Hairer, Gerhard Wanner, and Syvert P Nørsett. *Solving ordinary differential equations I: Nonstiff problems*. Springer, 1993.
- [54] Charlotte Bay Hasager, Leif Rasmussen, Alfredo Peña, Leo E Jensen, and Pierre-Elouan Réthoré. “Wind farm wake: The Horns Rev photo case.” In: *Energies* 6.2 (2013), pp. 696–716.
- [55] Jan S. Hesthaven, Benjamin Peherstorfer, and Benjamin Unger. “Nonlinear model reduction for transport-dominated problems.” In: *Acta Numerica* 35 (2026), 173–272. DOI: [10.1017/S0962492926100294](https://doi.org/10.1017/S0962492926100294).
- [56] Jan S Hesthaven, Gianluigi Rozza, Benjamin Stamm, et al. *Certified reduced basis methods for parametrized partial differential equations*. Vol. 590. Springer, 2016.
- [57] Michael Hinze and Stefan Volkwein. “Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control.” In: *Dimension Reduction of Large-Scale Systems: Proceedings of a Workshop held in Oberwolfach, Germany, October 19–25, 2003*. Springer. 2005, pp. 261–306.
- [58] Angelo Iollo, Stéphane Lanteri, and Jean-Antoine Désidéri. “Stability properties of POD–Galerkin approximations for the compressible Navier–Stokes equations.” In: *Theoretical and Computational Fluid Dynamics* 13.6 (2000), pp. 377–396.
- [59] Kazufumi Ito and Sivaguru S Ravindran. “A reduced-order method for simulation and control of fluid flows.” In: *Journal of Computational Physics* 143.2 (1998), pp. 403–425.
- [60] Seunghyon Kang, Hyeonhuh Kim, and Boris Kramer. “Parametric Operator Inference to Simulate the Purging Process in Semiconductor Manufacturing.” In: *IEEE Transactions on Semiconductor Manufacturing* (2026).
- [61] Robin Klein. “Structure-Preserving Hyper-Reduction Methods for the Incompressible Navier-Stokes Equations.” MA thesis. Delft University of Technology, 2022.
- [62] Tomoki Koike and Elizabeth Qian. “Energy-preserving reduced operator inference for efficient design and Control.” In: *AIAA SCITECH 2024 Forum* (2024). DOI: [10.2514/6.2024-1012](https://doi.org/10.2514/6.2024-1012).

- [63] Boris Kramer, Benjamin Peherstorfer, and Karen E Willcox. "Learning nonlinear reduced models from data with operator inference." In: *Annual Review of Fluid Mechanics* 56.1 (2024), pp. 521–548.
- [64] Boris Kramer and Elizabeth Qian. "Reduced-order modeling for engineering systems: survey and opportunities for digital twins: B. Kramer, E. Qian." In: *Structural and Multidisciplinary Optimization* 69.6 (2026), p. 141.
- [65] J Nathan Kutz, Steven L Brunton, Bingni W Brunton, and Joshua L Proctor. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.
- [66] Sanjay Lall, Petr Krysl, and Jerrold E Marsden. "Structure-preserving model reduction for mechanical systems." In: *Physica D: Nonlinear Phenomena* 184.1-4 (2003), pp. 304–318.
- [67] C Lieberman, K Fidkowski, K Willcox, and B van Bloemen Waanders. "Hessian-based model reduction: large-scale inversion and prediction." In: *International Journal for Numerical Methods in Fluids* 71.2 (2013), pp. 135–150.
- [68] Babak Maboudi Afkham and Jan S Hesthaven. "Structure-preserving model-reduction of dissipative Hamiltonian systems." In: *Journal of Scientific Computing* 81.1 (2019), pp. 3–21.
- [69] Shane A McQuarrie, Cheng Huang, and Karen E Willcox. "Data-driven reduced-order models via regularised operator inference for a single-injector combustion process." In: *Journal of the Royal Society of New Zealand* 51.2 (2021), pp. 194–211.
- [70] Shane A McQuarrie, Parisa Khodabakhshi, and Karen E Willcox. "Nonintrusive reduced-order models for parametric partial differential equations via data-driven operator inference." In: *SIAM Journal on Scientific Computing* 45.4 (2023), A1917–A1946.
- [71] Muhammad Mohebujjaman, Leo G Rebholz, Xuping Xie, and Traian Iliescu. "Energy balance and mass conservation in reduced order models of fluid flows." In: *Journal of Computational Physics* 346 (2017), pp. 262–277.
- [72] Parviz Moin and Krishnan Mahesh. "Direct numerical simulation: a tool in turbulence research." In: *Annual Review of Fluid Mechanics* 30.1 (1998), pp. 539–578.
- [73] RA Nicolaides. "n a Class of Finite Elements Generated by Lagrange Interpolation." In: *SIAM Journal on Numerical Analysis* 9.3 (1972), pp. 435–445.

- [74] Bernd R Noack, Paul Papas, and Peter A Monkewitz. "The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows." In: *Journal of Fluid Mechanics* 523 (2005), pp. 339–365.
- [75] Alberto Padovan, Blaine Vollmer, and Daniel J Bodony. "Data-driven model reduction via non-intrusive optimization of projection operators and reduced-order dynamics." In: *SIAM Journal on Applied Dynamical Systems* 23.4 (2024), pp. 3052–3076.
- [76] Eric J Parish and Francesco Rizzi. "On the impact of dimensionally-consistent and physics-based inner products for POD-Galerkin and least-squares model reduction of compressible flows." In: *Journal of Computational Physics* 491 (2023), p. 112387.
- [77] Benjamin Peherstorfer. "Sampling low-dimensional Markovian dynamics for preasymptotically recovering reduced models from data with operator inference." In: *SIAM Journal on Scientific Computing* 42.5 (2020), A3489–A3515.
- [78] Benjamin Peherstorfer and Karen Willcox. "Data-driven operator inference for nonintrusive projection-based model reduction." In: *Computer Methods in Applied Mechanics and Engineering* 306 (2016), pp. 196–215. ISSN: 0045-7825. DOI: <https://doi.org/10.1016/j.cma.2016.03.025>. URL: <https://www.sciencedirect.com/science/article/pii/S0045782516301104>.
- [79] Liqian Peng and Kamran Mohseni. "Symplectic model reduction of Hamiltonian systems." In: *SIAM Journal on Scientific Computing* 38.1 (2016), A1–A27.
- [80] Janet S Peterson. "The reduced basis method for incompressible viscous flow calculations." In: *SIAM Journal on Scientific and Statistical Computing* 10.4 (1989), pp. 777–786.
- [81] Ugo Piomelli. "Large-eddy simulation: achievements and challenges." In: *Progress in Aerospace Sciences* 35.4 (1999), pp. 335–362.
- [82] Josep Plana-Riu, Henrik Rosenberger, Benjamin Sanderse, and F. Xavier Trias. "Stable self-adaptive timestepping for Reduced Order Models for incompressible flows." In: *Journal of Computational Physics* (2026), p. 115104. DOI: [10.1016/j.jcp.2026.115104](https://doi.org/10.1016/j.jcp.2026.115104).
- [83] Josep Plana-Riu, Henrik Rosenberger, Benjamin Sanderse, and F Xavier Trias. *Stable self-adaptive timestepping for Reduced Order Models for incompressible flows*. 2026. DOI: [10.1016/j.jcp.2026.115104](https://doi.org/10.1016/j.jcp.2026.115104).

- [84] Stephen Prajna. "POD model reduction with stability guarantee." In: *42nd IEEE International Conference on Decision and Control (IEEE Cat. No. 03CH37475)*. Vol. 5. IEEE. 2003, pp. 5254–5258.
- [85] Christophe Prud'Homme, Dimitrios V Rovas, Karen Veroy, Luc Machiels, Yvon Maday, Anthony T Patera, and Gabriel Turinici. "Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods." In: *J. Fluids Eng.* 124.1 (2002), pp. 70–80.
- [86] Elizabeth Qian, Boris Kramer, Benjamin Peherstorfer, and Karen Willcox. "Lift & learn: Physics-informed machine learning for large-scale nonlinear dynamical systems." In: *Physica D: Nonlinear Phenomena* 406 (2020), p. 132401.
- [87] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." In: *Journal of Computational physics* 378 (2019), pp. 686–707.
- [88] Henrik Rosenberger. "Advances in Structure-Preserving Reduced Order Models for the Incompressible Navier-Stokes Equations." MA thesis. Eindhoven University of Technology, 2021.
- [89] Henrik Rosenberger and Benjamin Sanderse. "Momentum-conserving ROMs for the incompressible Navier-Stokes equations." In: *8th European Congress on Computational Methods in Applied Sciences and Engineering* (2022). DOI: [10.23967/eccomas.2022.186](https://doi.org/10.23967/eccomas.2022.186).
- [90] Henrik Rosenberger and Benjamin Sanderse. "No pressure? Energy-consistent ROMs for the incompressible Navier-Stokes equations with time-dependent boundary conditions." In: *Journal of Computational Physics* 491 (2023), p. 112405. DOI: [10.1016/j.jcp.2023.112405](https://doi.org/10.1016/j.jcp.2023.112405).
- [91] Henrik Rosenberger, Benjamin Sanderse, and Giovanni Stabile. "Exact operator inference decouples parametric problems." In: *Computer Methods in Applied Mechanics and Engineering* 461 (2026), p. 119199. DOI: [10.1016/j.cma.2026.119199](https://doi.org/10.1016/j.cma.2026.119199).
- [92] Henrik Rosenberger, Benjamin Sanderse, and Giovanni Stabile. "Exact operator inference with minimal data." In: *SIAM Journal on Scientific Computing* 48.5 (2026), A2391–C1160. DOI: [10.1137/25M177248](https://doi.org/10.1137/25M177248).
- [93] Clarence W Rowley, Tim Colonius, and Richard M Murray. "Model reduction for compressible flows using POD and Galerkin projection." In: *Physica D: Nonlinear Phenomena* 189.1-2 (2004), pp. 115–129.

- [94] Gianluigi Rozza, DB Huynh, and Andrea Manzoni. "Reduced basis approximation and a posteriori error estimation for Stokes flows in parametrized geometries: roles of the inf-sup stability constants." In: *Numerische Mathematik* 125.1 (2013), pp. 115–152.
- [95] Gianluigi Rozza, Dinh Bao Phuong Huynh, and Anthony T Patera. "Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations: application to transport and continuum mechanics." In: *Archives of Computational Methods in Engineering* 15.3 (2008), pp. 229–275.
- [96] Pierre Sagaut. *Large eddy simulation for incompressible flows: an introduction*. Springer, 2006.
- [97] Benjamin Sanderse. "Energy-conserving discretization methods for the incompressible Navier-Stokes equations: application to the simulation of wind-turbine wakes." PhD thesis. Eindhoven University of Technology, 2013.
- [98] Benjamin Sanderse. "Non-linearly stable reduced-order models for incompressible flow with energy-conserving finite volume methods." In: *Journal of Computational Physics* 421 (2020), p. 109736.
- [99] Benjamin Sanderse and Barry Koren. "Accuracy analysis of explicit Runge–Kutta methods applied to the incompressible Navier–Stokes equations." In: *Journal of Computational Physics* 231.8 (2012), pp. 3041–3063.
- [100] Kamron Saniee. "A simple expression for multivariate Lagrange interpolation." In: *SIAM Undergraduate Research Online* 1.1 (2008), pp. 1–9. URL: <https://doi.org/10.1137/08S010025>.
- [101] Nihar Sawant, Boris Kramer, and Benjamin Peherstorfer. "Physics-informed regularization and structure preservation for learning stable reduced models from data with operator inference." In: *Computer Methods in Applied Mechanics and Engineering* 404 (2023), p. 115836.
- [102] Gilles Serre, Philippe Lafon, Xavier Gloerfelt, and Christophe Bailly. "Reliable reduced-order models for time-dependent linearized Euler equations." In: *Journal of Computational Physics* 231.15 (2012), pp. 5176–5194.
- [103] Harsh Sharma, Zhu Wang, and Boris Kramer. "Hamiltonian operator inference: Physics-preserving learning of reduced-order models for canonical Hamiltonian systems." In: *Physica D: Non-linear Phenomena* 431 (2022), p. 133122.

- [104] K Chad Sockwell. “Mass Conserving Hamiltonian-Structure-Preserving Reduced Order Modeling for the Rotating Shallow Water Equations Discretized by a Mimetic Spatial Scheme.” PhD thesis. The Florida State University, 2019.
- [105] Giovanni Stabile and Gianluigi Rozza. “Finite volume POD-Galerkin stabilised reduced order methods for the parametrised incompressible Navier–Stokes equations.” In: *Computers & Fluids* 173 (2018), pp. 273–284.
- [106] Renee Swischuk, Boris Kramer, Cheng Huang, and Karen Willcox. “Learning physics-based reduced-order models for a single-injector combustion process.” In: *AIAA Journal* 58.6 (2020), pp. 2658–2672.
- [107] Lloyd N Trefethen. *Approximation theory and approximation practice, extended edition*. SIAM, 2019.
- [108] Francesc Xavier Trias, Oriol Lehmkuhl, Assensi Oliva, Carles David Pérez-Segarra, and RWCP Verstappen. “Symmetry-preserving discretization of Navier–Stokes equations on collocated unstructured grids.” In: *Journal of Computational Physics* 258 (2014), pp. 246–267.
- [109] Karsten Urban and Anthony T Patera. “A new error bound for reduced basis approximation of parabolic partial differential equations.” In: *Comptes Rendus. Mathématique* 350.3-4 (2012), pp. 203–207.
- [110] Wayne Isaac Tan Uy, Yuepeng Wang, Yuxiao Wen, and Benjamin Peherstorfer. “Active operator inference for learning low-dimensional dynamical-system models from noisy data.” In: *SIAM Journal on Scientific Computing* 45.4 (2023), A1462–A1490.
- [111] Karen Veroy and Anthony T Patera. “Certified real-time solution of the parametrized steady incompressible Navier–Stokes equations: rigorous reduced-basis a posteriori error bounds.” In: *International Journal for Numerical Methods in Fluids* 47.8-9 (2005), pp. 773–788.
- [112] RWCP Verstappen and AEP Veldman. “Symmetry-preserving discretization of turbulent flow.” In: *Journal of Computational Physics* 187.1 (2003), pp. 343–368.
- [113] Arjun Vijaywargiya, Shane A McQuarrie, and Anthony Gruber. “Tensor parametric Hamiltonian operator inference.” In: *arXiv preprint arXiv:2502.10888* (2025).

- [114] Stefan Volkwein. "Model reduction using proper orthogonal decomposition." In: *Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz*. see <http://www.uni-graz.at/imawww/volkwein/POD.pdf> 1025 (2011).
- [115] Zhu Wang, Imran Akhtar, Jeff Borggaard, and Traian Iliescu. "Proper orthogonal decomposition closure models for turbulent flows: a numerical comparison." In: *Computer Methods in Applied Mechanics and Engineering* 237 (2012), pp. 10–26.
- [116] David C Wilcox et al. *Turbulence modeling for CFD*. Vol. 2. DCW industries La Canada, CA, 1998.
- [117] Christopher K. I. Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA, 2006.
- [118] Manyu Xiao, Piotr Breitzkopf, Rajan Filomeno Coelho, Catherine Knopf-Lenoir, Pierre Villon, and Weihong Zhang. "Constrained proper orthogonal decomposition based on QR-factorization for aerodynamical shape optimization." In: *Applied Mathematics and Computation* 223 (2013), pp. 254–263.
- [119] Süleyman Yıldız, Pawan Goyal, Peter Benner, and Bülent Karasözen. "Learning reduced-order dynamics for parametrized shallow water equations from data." In: *International Journal for Numerical Methods in Fluids* 93.8 (2021), pp. 2803–2821.
- [120] Olgierd Cecil Zienkiewicz, Robert Leroy Taylor, Perumal Nithiarasu, and JZ Zhu. *The finite element method*. Vol. 3. Elsevier, 1977.

LIST OF PUBLICATIONS

Publications presented in this thesis:

- Henrik Rosenberger and Benjamin Sanderse. “No pressure? Energy-consistent ROMs for the incompressible Navier-Stokes equations with time-dependent boundary conditions.” In: *Journal of Computational Physics* 491 (2023), p. 112405. DOI: [10.1016/j.jcp.2023.112405](https://doi.org/10.1016/j.jcp.2023.112405)
- Henrik Rosenberger, Benjamin Sanderse, and Giovanni Stabile. “Exact operator inference with minimal data.” In: *SIAM Journal on Scientific Computing* 48.5 (2026), A2391–C1160. DOI: [10.1137/25M177248](https://doi.org/10.1137/25M177248)
- Henrik Rosenberger, Benjamin Sanderse, and Giovanni Stabile. “Exact operator inference decouples parametric problems.” In: *Computer Methods in Applied Mechanics and Engineering* 461 (2026), p. 119199. DOI: [10.1016/j.cma.2026.119199](https://doi.org/10.1016/j.cma.2026.119199).

Other publications:

- Henrik Rosenberger and Benjamin Sanderse. “Momentum-conserving ROMs for the incompressible Navier-Stokes equations.” In: *8th European Congress on Computational Methods in Applied Sciences and Engineering* (2022). DOI: [10.23967/eccomas.2022.186](https://doi.org/10.23967/eccomas.2022.186)
- Krishan Chand, Henrik Rosenberger, and Benjamin Sanderse. “A pressure-free long-time stable reduced-order model for two-dimensional Rayleigh–Bénard convection.” In: *Chaos: An Interdisciplinary Journal of Nonlinear Science* 34.2 (2024). DOI: [10.1063/5.0168857](https://doi.org/10.1063/5.0168857)
- Josep Plana-Riu, Henrik Rosenberger, Benjamin Sanderse, and F. Xavier Trias. “Stable self-adaptive timestepping for Reduced Order Models for incompressible flows.” In: *Journal of Computational Physics* (2026), p. 115104. DOI: [10.1016/j.jcp.2026.115104](https://doi.org/10.1016/j.jcp.2026.115104)

ABOUT THE AUTHOR

Henrik Karl Erich Rosenberger was born in Rottweil, Germany, on October 5, 1997. In 2016, he graduated from secondary school Leibniz-Gymnasium Rottweil and went on to study Simulation Technology at University of Stuttgart, Germany. In 2019, he obtained his Bachelor of Science with honors and continued his studies within a Double Degree program that combined a Master of Science in Simulation Technology at University of Stuttgart and a Master of Science in Industrial and Applied Mathematics at Eindhoven University of Technology. In 2021, he graduated with honors on a thesis titled 'Advances in Structure-Preserving Reduced Order Models for the Incompressible Navier-Stokes Equations' under supervision of prof. dr. ir. B. Koren and dr. ir. B. Sanderse. From 2021 to 2026, he performed PhD research in the Scientific computing group of dr. ir. B. Sanderse at Centrum Wiskunde & Informatica, Amsterdam.

ACKNOWLEDGEMENTS

I want to thank my supervisors Benjamin and Giovanni for their guidance and feedback throughout this project. I have benefitted from every single of our regular meetings.

Many thanks to my colleagues in the Scientific Computing group for the always welcoming atmosphere. It was a pleasure to work and especially to attend conferences with you. Thanks to all people that made me feel home at CWI, in academia and in Amsterdam, in particular my flatmates.

I also want to thank my family and friends in Germany for visiting me in Amsterdam, hosting me when in Germany and offering distraction and outside perspectives whenever needed.

Finally, I want to thank Michiel for all the support in the end phase of my PhD.

SUMMARY

Computational efficiency is of paramount importance in complex engineering tasks such as optimization, data assimilation, uncertainty quantification and the development of digital twins. All these tasks require multiple simulations with similar settings. While classical numerical methods such as Finite Elements have primarily been developed to run single simulations, data-driven methods are being developed to exploit the similarity among such multiple simulations.

One of these data-driven methods is projection-based model reduction. Like classical numerical methods, this method derives discrete models from first principles. However, while classical numerical methods result in generic discretizations, projection-based model reduction leverages data to construct more efficient, problem-specific discretizations.

This thesis comprises three advances of projection-based model reduction. The first advance is the application to the incompressible Navier-Stokes equations with time-dependent boundary conditions. This application results in a reduced-order model (ROM) that does not require the computation of the pressure to simulate the velocity and preserves the structure of the kinetic energy evolution. It is shown that this velocity-only ROM is equivalent to a velocity-pressure ROM, i.e., a ROM that simulates both velocity and pressure. The advantages of the velocity-only ROM are better stability and computational efficiency.

The second advance is the proposition of a method that enables the construction of projection-based ROMs non-intrusively, i.e., without access to the corresponding full-order model. Such access is often not possible with software used in industry, so operator inference has been introduced for nonintrusive ROM construction. However, operator inference reconstructs the corresponding intrusive ROM only if sufficiently informative snapshot data is available. The novel method generates such data and thereby guarantees the exact reconstruction. This exact operator inference is optimal in the sense that the amount of generated data to achieve exact reconstruction is minimal.

The third advance is the application of exact operator inference to parametric models. Next to the exact reconstruction with minimal data, this application allows to decouple the parametric operator inference minimization problem. The resulting decoupled problems are embarrassingly parallel, numerically more stable and can be solved more efficiently.

The main result of this thesis is the proposed exact operator inference that enables the exact non-intrusive reconstruction of intrusive ROMs. This method can be applied to many scientific and engineering problems, including the incompressible Navier-Stokes equations as studied in this thesis.