Advanced Techniques for Dimensionality Reduction of Nonlinear and Parametric Dynamical Systems using System-Theoretic Measures

A thesis
submitted by

Danish Rafiq
2017PHAELE006

for the award of the degree of

Doctor of Philosophy

in

Electrical Engineering

Advisor: Dr. Mohammad Abid Bazaz

Department of Electrical Engineering
National Institute of Technology Srinagar

July 2022

Thesis Certificate

This is to certify that the thesis entitled **Advanced Techniques for Dimensionality Reduction of Nonlinear and Parametric Dynamical Systems using System-Theoretic Measures**, submitted by **Danish Rafiq** bearing Enrollment Number **2017PHAELE006** to the National Institute of Technology Srinagar for the award of the degree of **Doctor of Philosophy**, is a bonafide record of the research work carried by him under my supervision. The contents of this thesis, in whole or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Mohammad Abid Bazaz

Associate Professor & Head
Electrical Engineering Department
National Institute of Technology Srinagar
190006

Thesis Certificate

This is to certify that the work contained in this thesis entitled **Advanced Techniques for Dimensionality Reduction of Nonlinear and Parametric Dynamical Systems using System-Theoretic Measures** is a bonafide research submitted to Department of Electrical Engineering, National Institute of Technology, is a bonafied work of **Danish Rafiq** bearing Enrollment No. **2017PHAELE006** and is worthy of consideration for the award of Doctor of Philosophy (Ph.D.) in Electrical Engineering.

Dr. Mohammad Abid Bazaz

Associate Professor & Head
Electrical Engineering Department
National Institute of Technology Srinagar
190006

Declaration

I hereby certify that the thesis entitled **Advanced Techniques for Dimensionality Reduction of Nonlinear and Parametric Dynamical Systems using System-Theoretic Measures**, is my own work and has been completed under the supervision of Dr. Mohammad Abid Bazaz (Associate Professor & Head, Department of Electrical Engineering, NIT Srinagar). To the best of my knowledge, it does not contain any material, written or published by another author, except where appropriate acknowledgment has been made in the text. This thesis has not been submitted to any other university or institute for the award of any degree or diploma.

Danish Rafiq
Enrollment Number: 2017PHAELE006
Electrical Engineering Department
National Institute of Technology Srinagar
190006

List of Publications

Journals

1. **D. Rafiq**, and M. A. Bazaz, A collection of large-scale benchmark models for nonlinear model order reduction, *Archives of Computational Methods in Engineering*, 2022.

2. **D. Rafiq**, J. Farooq, and M. A. Bazaz, Synergistic use of intrusive and non-intrusive model order reduction techniques for dynamical power grids, *International Journal of Electrical Power and Energy Systems*, **138**:107908, 2022.
https://www.sciencedirect.com/science/article/abs/pii/S0142061521011194

3. **D. Rafiq**, and M. A. Bazaz, Model order reduction via moment-matching: A state-of-the-art review, *Archives of Computational Methods in Engineering*, **29**:1463-1483,2022.
https://link.springer.com/article/10.1007/s11831-021-09618-2

4. **D. Rafiq**, and M. A. Bazaz, Adaptive parametric sampling scheme for nonlinear model order reduction, *Nonlinear Dynamics,* **107**:813-828, 2021.
https://doi.org/10.1007/s11071-021-07025-7.

5. **D. Rafiq**, and M. A. Bazaz, Nonlinear Model Order Reduction via Nonlinear Moment Matching with Dynamic Mode Decomposition, *International Journal of Nonlinear Mechanics*, **128**:103625, 2021.
https://www.sciencedirect.com/science/article/abs/pii/S0020746220302870

6. **D. Rafiq**, and M. A. Bazaz, Efficient computation of 1D and 2D nonlinear viscous Burgers' equation, *Journal of Dynamics & Control*, **9**:1523-1535, 2021.
https://link.springer.com/article/10.1007/s40435-021-00772-w

7. **D. Rafiq**, and M. A. Bazaz, A framework for parametric reduction in large-scale nonlinear dynamical systems, *Nonlinear Dynamics*, **102**:1897-1908, 2020.
https://link.springer.com/article/10.1007/s11071-020-05970-3

8. **D. Rafiq**, and M. A. Bazaz, A comprehensive scheme for reduction of non-linear dynamical systems, *International Journal of Dynamics & Control* , **8**:361-369, 2020.
https://link.springer.com/article/10.1007/s40435-019-00589-8

9. **D. Rafiq**, A. Batool, and M. A. Bazaz, Three months of COVID-19: A systematic review and meta-analysis, *Reviews in Medical Virology*, **30**:e2113, 2020.
https://onlinelibrary.wiley.com/doi/10.1002/rmv.2113

10. **D. Rafiq**, S. A. Suhail, and M. A. Bazaz, Evaluation and prediction of COVID-19 in India: A case study of worst hit states, *Chaos, Solitons & Fractals*, **139**:110014, 2020.
https://www.sciencedirect.com/science/article/pii/S0960077920304124

Referred Conferences

1. A. Hamid, **D. Rafiq**, S. A. Nahvi, and M. A. Bazaz, Power Grid parameter estimation using sparse Identification of nonlinear dynamics, *International Conference on Intelligent Controller and Computing Smart Power*, 2022.

2. A. Hamid, **D. Rafiq**, S. A. Nahvi, and M. A. Bazaz, Discovering low-rank representations of large-scale power-grid models using Koopman theory, *IEEE 2022 Trends in Electrical, Electronics, Computer Engineering Conference (TEECCON)*, 2022

3. **D. Rafiq**, and M. A. Bazaz, Structure preserving nonlinear reduced order modeling technique for power systems, *Seventh Indian Control Conference (ICC)*, 2021.

4. **D. Rafiq**, and M. A. Bazaz, A comprehensive scheme for fast simulation of Burgers' equation, *Sixth Indian Control Conference (ICC)*, IEEE 397-402, 2019.

5. **D. Rafiq**, and M. A. Bazaz, Model order reduction of non-linear transmission lines using non-linear moment matching, *2019 International Conference on Computing, Power and Communication Technologies (GUCON)*, 394-399, 2019.

Abstract

High-fidelity modeling and simulation has become indispensable across many engineering applications as it provides physically realistic responses to engineered systems that would otherwise require costly experiments. However, as we rely more and more on such simulations, we must imbue them with very high levels of fidelity to make decisions with them. The model order reduction (MOR) technique has garnered attention since it has enabled the feasibility of rapid simulations, real-time assessment, and control.

This thesis studies reduction in nonlinear dynamical systems via system theoretic methods. Notably, we seek to replace large-scale discretized models emerging from the time-dependent nonlinear partial differential equations (PDEs) with computationally cheaper models crafted to control the loss of fidelity during reduction. We start by introducing the problem pervasive in large-scale modeling regimes. We build upon the idea of the existing generalization of moment-matching to nonlinear systems based on steady-state considerations. A connection between the linear and nonlinear enhancements of moment-matching allows extracting reduced models in a snapshot-free architecture that drastically reduces the complexity bottleneck in nonlinear MOR. Towards this direction, we propose some efficient reduction methods for nonlinear state-space models based on approximate moment-matching. The resulting schemes are extensively discussed and tested on various benchmark nonlinear models. We also discuss the application of proposed frameworks to nonlinear second-order systems.

In the realm of parameterized nonlinear PDEs, where the model parameter guarantees the existence of several distinct dynamical regimes, we capitalize on the fact that smooth parametric dependence allows building low-rank approximation models that capture the true response over a wide range of parameters via interpolation. This idea is motivated by design-based strategies prevalent in the industry; the cost associated with constructing ROMs for different parameter values should be amortized by reusing them in the interpolation process. We demonstrate this idea through several benchmark examples.

Towards the end of the thesis, we focus on adaptive parameter sampling strategies and show how subspace angles could be used to extract meaningful information for remeshing the parametric domain of interest. This is again demonstrated on various examples from circuits and MEMS. We also provide some source codes/simulation files to promote transparency and reproducibility.

Acknowledgments

First and foremost, thanks (*Alhamdulillah*) to Allah (*S.W.T*) for granting me health and patience throughout the years, particularly during the times of the pandemic. My greatest thanks to my advisor, Dr. Mohammad Abid Bazaz, who has been both a role model and a source of encouragement and guidance throughout my Ph.D. He has provided a supportive environment that has allowed me to explore my interests, and I am grateful for his support and optimism about my work. I would like to extend my warm regards to Prof. Jean-Mathieu Mencik from INSA Centre Val dé Loire, France and Prof. Madhu N. Belur from IIT Bombay, India for taking time to review my thesis. Their valuable suggestions really improved the presentation of this thesis. I am also thankful to my former advisor Dr. Shahkar A. Nahvi for his words of encouragement and motivation.

I am thankful to the Department of Electrical Engineering faculty, especially Prof. A. H. Bhat for chairing the examination committee and Dr. Neeraj Gupta, for his interest in my work and his words of appreciation. I want to acknowledge Dr. Maria Cruz Varona for sharing the benchmark models and fruitful discussions on NLMM. She has helped me a lot in my initial years. I am much obliged to Prof. Alessandro Alla for our valuable discussions on DMD.

Working in Dr. Bazaz's group has indeed been a pleasure. I especially enjoyed the time spent with current and former lab mates, who have taught me a great deal: Hadhiq Khan, Satyavir Singh, Suhail Ahmad Suhail, Zahid Afzal Thoker, Abdul Waheed, Masood Nazir, Viqar Yousuf, Ahmad Suhail, Faisal Jamsheed, Junaid Farooq, Faizan Hassan, and others. Dr. Hadhiq has been incredibly kind to me, and I have learned a lot from him in one way or the other.

Thanks to my loving parents, my first and greatest teachers, and my dear sister Dr. Asiya Batool, her love and support have not wavered. I would not be where I am without the love and support of my family. Moreover, I am grateful to my relatives and friends for providing comforting distraction from research.

Lastly, I gratefully acknowledge the support of this work from the Ministry of Human Resouce, India, for my Doctoral fellowship.

Contents

Glossary

Dimensions

m	Dimension of the input variable $\mathbf{u} \in \mathbb{R}^m$
n	Dimension of the state $\mathbf{x} \in \mathbb{R}^n$
p	Dimension of the measurement or output variable $\mathbf{y} \in \mathbb{R}^p$
q	Dimension of parametric space $\boldsymbol{\mu} \in \mathbb{R}^q$
r	Dimension of the reduced state $\mathbf{x}_r \in \mathbb{R}^r$

Scalars

h	Time step
λ	Eigen value
L	Spatial length
m_d	Number of dominant Singular Values in DEIM
m_D	Reduced rank representation for DMD
σ	Interpolation point
n_s	Number of temporal snapshots in POD
s	Laplace variable
t	Time
T	Final time
x	Spatial variable
$\theta_{\max}$	Maximum tolerance for subspace angles

Vectors

$\mathbf{b}$	Input vector
$\mathbf{c}$	Output vector
$\mathbf{e}$	Error vector
$\mathbf{r}$	Residual vector
$\mathbf{u}$	Vector of inputs
$\mathbf{v}$	Left tangential direction
$\mathbf{w}$	Right tangential direction
$\mathbf{x}$	State of a system

Vectors, continued

$\mathbf{x}_0$	Initial condition on $\mathbf{x}$
$\mathbf{x}_r$	Reduced state
$\mathbf{y}$	Vector of measurements
$\boldsymbol{\zeta}$	State of Signal Generator
$\boldsymbol{\eta}_i$	Moment at i

Matrices

$(\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C})$	Matrices for continuous-time state-space system
$(\mathbf{E}_r, \mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r)$	Matrices for reduced state-space system
$\mathbf{F}$	Matrix of nonlinear snapshots for DEIM $\mathbf{F} \in \mathbb{R}^{n \times n_s}$
$\mathbf{h}$	Impulse response matrix $\mathbf{h} \in \mathbb{R}^{p \times m}$
$\mathbf{J}$	Jacobian Matrix
$(\mathbf{M}, \mathbf{D}, \mathbf{K})$	Matrices for continuous-time second-order state-space system
$(\mathbf{M}_r, \mathbf{D}_r, \mathbf{K}_r)$	Matrices for reduced second-order state-space system
$\mathcal{O}$	Observability Gramian matrix
$\mathcal{P}$	Reachability Gramian matrix
$\mathbf{P}$	Projection matrix
$\mathbf{U}_\mathrm{d}$	DEIM basis $\mathbf{U}_\mathrm{d} \in \mathbb{R}^{n \times m_d}$
$\mathbf{V}$	Left projection matrix
$\mathbf{V}_\mathrm{POD}$	POD basis $\mathbf{V}_\mathrm{POD} \in \mathbb{R}^{n \times r}$
$\mathbf{V}_\mathrm{NLMM}$	NLMM basis $\mathbf{V}_\mathrm{NLMM} \in \mathbb{R}^{n \times r}$
$\mathbf{V}_\mathrm{DMD}$	DMD modes $\mathbf{V}_\mathrm{d} \in \mathbb{R}^{n \times m_d}$
$\mathbf{W}$	Right Projection matrix
$\boldsymbol{\Sigma}$	Matrix of singular values
$\mathbf{X}$	Snapshot data matrix $\mathbf{X} \in \mathbb{C}^{n \times n_s}$
$(\mathbf{X_1}, \mathbf{X_2})$	Data matrices for DMD
$(\boldsymbol{\Xi}, \boldsymbol{\Psi})$	Matrices for linear signal generator
$\boldsymbol{\Lambda}$	Diagonal matrix of eigen-values
$\boldsymbol{\Upsilon}$	DMD eigen-vectors

Operators, functions, and maps

$\mathbf{f}$	Nonlinear vector-valued function $\mathbf{f} \in \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$
$\mathbf{g}$	Vector-valued measurement functions on $\mathbf{x}$
$\mathbf{G}(s)$	Transfer function matrix $\mathbf{G}(s) \in \mathbb{R}^{p \times m}$
$\wp$	Nonlinear Petrov-Galerkin mapping $\wp :\in \mathbb{R}^r \to \mathbb{R}^n$
(ϖ, ϱ)	Nonlinear mappings of signal-generator
$\mathcal{K}$	Krylov subspace
$\mathcal{V}$	r-dimensional subspace spanned by columns of $\mathbf{V}$
$\mathcal{W}$	r-dimensional subspace spanned by columns of $\mathbf{W}$

Operators, functions, and maps, continued

$\boldsymbol{\varphi}$	Nonlinear reduction mapping $\boldsymbol{\varphi} \in \mathbb{R}^n \to \mathbb{R}^r$
$\otimes$	Knocker product
$\dagger$	Moore-Penrose pseudoinverse
$\boldsymbol{\mu}$	Parameter of interest
$\mathcal{N}$	Columnspace of snapshot matrix $\mathbf{X}$
$\langle ., . \rangle$	Inner product

Norms

$\|.\|_0$	$\mathcal{L}_0$ pseduo-norm of vector $\mathbf{x}$
$\|.\|_1$	$\mathcal{L}_1$ norm of vector $\mathbf{x}$ given by $\|\mathbf{x}\|_1 = \sum_{i=1}^{n} \|\mathbf{x}_i\|$
$\|.\|_2$	$\mathcal{L}_2$ norm of vector $\mathbf{x}$ given by $\|\mathbf{x}\|_2 = \sqrt{\sum_{i=1}^{n} \|(\mathbf{x}_i)^2\|}$
$\|.\|_2$	2-norm of matrix $\mathbf{X}$ given by $\|\mathbf{X}\|_2 = \max_x \frac{\|\mathbf{X}x\|_2}{\|x\|_2} \quad (\|x\| \neq 0)$
$\|.\|_F$	Frobenius-norm of matrix $\mathbf{X}$ given by $\|.\|_F = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{m} \|\mathbf{X}_{ij}\|^2}$

Most common Acronyms

DEIM	Discrete Empirical Interpolation Method
DMD	Dynamic Mode Decomposition
FDM	Finite Difference Method
FEM	Finite Element Method
FOM	Full Order Model
FVM	Finite Volume Method
LSS	Large-Scale Systems
LTI	Linear Time-Invariant
MOR	Model Order Reduction
MEMS	Micro-Electro-Mechanical System
MIMO	Multiple-Input Multiple Output
NLMM	Nonlinear Moment Matching
NLSE	Nonlinear System of Equations
NMOR	Nonlinear Model Order Reduction
ODE	Ordinary Differential Equation
PDE	Partial Differential Equation
PMOR	Parametric Model Order Reduction
POD	Proper Orthogonal Decomposition
ROM	Reduced Order Model
SO-NLMM	Second-Order Nonlinear Moment Matching
SISO	Single-Input Single Output
SVD	Singular Value Decomposition
TPWL	Trajectory Piece-wise Linear

List of Statements

List of Figures

List of Algorithms

List of Tables

Part I

Preliminaries

Chapter 1

Introduction and Rationale of the study

1.1 Motivation for reduced-order modeling paradigm

Understanding physical and artificial processes is of fundamental interest among all branches of engineering and contemporary sciences. While the governing physical laws of such models are mainly described by mathematical equations, the model complexity remains a central challenge. Among them is the dimension of the problem. The "large-scale" nature of dynamical systems often leads to the over-burdening of computational resources and can become impractical at times. Large scale systems (LSS) arise in various technical systems having a large number of individual subsystems or components, such as in integrated circuits, VLSI systems [201, 291, 197], gas and grid networks. Furthermore, large-scale models also appear as a result of the mathematical treatment of various engineered systems. A good example is the spatial discretization of partial differential equations (PDEs) describing the underlying physics. A fine discretization over the 2D/3D geometric domain via the finite-difference method (FDM), finite element method (FEM), or the finite-volume method (FVM) can lead to a model of dimension $n \approx 10^6$. This is a common practice in computational mechanics or fluid mechanics [246]. In addition, disease modeling, weather predictions, biological models, micro-electro-mechanical systems (MEMS), chemical processes, computational aerodynamics, neuroscience, cardiovascular systems, air-quality data assimilation, molecular dynamic systems, optimal cooling, etc., represent areas where large-scale models are pervasive.

During a simulation, one intends to predict or analyze the behavior of a system under study. As far as large-scale models are concerned, their simulation is not feasible as thousands or even millions of degrees of freedom are involved. Using conventional numerical techniques to simulate such systems involves thousands of core hours which can incur substantial computational costs. Highly accurate solutions demand even more distinguished resolution discretization methods, which

further compounds the complexity of these models. Furthermore, various engineering applications require repeated simulations for different design parameters of the system. This can very well become problematic if large-scale models are involved and hence prevent fast-turnaround designs.

Similar issues occur in optimization and control. In control, one seeks to modify the system's behavior to meet certain desired performances specifications. Such modification involves interconnecting the original model with a "controller" to meet the control design. Generically, the complexity of the controller is approximately the same as that of the original plant. If the latter is too complex, it increases the potential burden on the controller in terms of storage, accuracy (due to ill-conditioning), and computational speed (due to limited bandwidth) [14].

Uncertainty quantification is another area where similar difficulties occur. Uncertainties can arise from model inputs, including physical parameters representing material properties, initial or boundary conditions. Various mathematical problems are encountered in solving forward or inverse uncertainty quantification problems involving high-dimensional coupled uncertainties and quickly makes the task intractable [81].

Apart from the issues mentioned above, a numerical constraint that has been there for a long time and is still a problem of concern is what is known as the *curse of dimensionality*. It refers to the problem wherein the space of possible parameter value sets grows exponentially with the number of unknown parameters.

All of the these factors have motivated researchers to go for reduced modeling strategies which have now become a de-facto tool to treat large-scale dynamical systems. Generally speaking, MOR methods aim to construct lower-order, parsimonious models that mimic the behavior of the original models while being computationally tractable to handle. This reduces the computational complexity and time consumed to solve large-scale engineering models. The sense of "dimension-reduction" or "complexity-reduction" stems from the idea that complex dynamical systems are often sparse in a relatively small subset of the whole space. Thus, if n is the size or dimension of the original problem, the solution space can be approximated by an r ($r << n$) dimensional subspace (or slow-manifold) where the sparse dynamics is embedded [14].

There are different approaches for reduced-order modeling. The first and foremost is to simplify the model geometry by representing the original model as a lumped-parameter model. This is achieved by making certain assumptions about the system or judiciously neglecting some effects. This technique, though is relatively simple, usually leads to an inaccurate reduced-order model (ROM). A different approach is to reduce the finite-element order (i.e., to use a coarser mesh), leading to a reduced set of equations by exploiting specific geometrical properties of the system. However, the speedup achieved is limited by the asymptotic range of convergence of the discretization method. A third option is to capture all physical effects of the original system by employing a fine discretization mesh and then apply MOR methods to full-order models (FOMs). In this thesis, we focus on the

third reduced modeling strategy.

The advantages of using a reduced model against the full model are apparent. In design optimizations and "what-if" scenarios, reduced models provide a considerable speedup of the simulation than the original model. This is especially beneficial in real-time applications such as control, digital twins, or predictive maintenance. In analog circuits, involving millions of transistors, ROMs can replace original sub-circuits to obtain a smaller set of equations and hence reduced computational effort. Reduced models are also used to estimate higher-level performance matrices of an extensive network. For example, device-level PDE simulations are carried out to capture the nonlinear behavior of transistors for aggressive memory designs. This often results in a large-scale system of ordinary-differential equations (ODEs) to solve. MOR techniques are then used to access circuit performances such as a memory cell's read/write access time. Furthermore, ROMs also help to better understand nonlinear phenomena. Nonlinear effects such as bifurcation or chaos can be applied in a low-dimensional framework to provide practical intuition of how large-scale nonlinear systems behave.

The following points highlight the common goals of reduced-order modeling techniques:

1. The reduced-model should very well approximate the response of the full-order model. This means that if the same input is applied to both the systems, the difference between the actual and approximated output responses should be *small* for a physically relevant norm over a wide range of inputs.

2. The cost of obtaining the reduced model should be less than performing the analysis via the full-order model.

3. Important system properties like stability and passivity should be preserved in reduced models.

4. Reduced order model should preserve critical system structures of the original model. This can include parameter dependence and second-order or time-delay structures.

5. Reduction schemes should be scalable, i.e., applicable to much larger systems, and provide an upper bound for the error.

6. Reduced modeling strategies should be robust and largely automatic to allow applicability in complex settings.

Although it is challenging to meet all the requirements mentioned above, especially in nonlinear dynamical systems, it is certainly possible to meet some.

1.2 Challenges in nonlinear model reduction

Most of the real-life and practical systems are nonlinear in nature. In electronic and circuit theory, nonlinearities arise from nonlinear resistors, transistors, diodes,

or magnetic inductors [14, 19]. The swing dynamics of alternators is nonlinear in nature. In mechanical applications, thin structures such as beams, plates, and shells undergoing large amplitude vibrations show geometric nonlinearities because of their relatively low bending stiffness. In fluid dynamics and heat transfer, nonlinear behavior is observed due to turbulence, convective acceleration, or radiation. Besides, nonlinear dynamics may arise in many technically relevant systems via nonlinear boundary conditions (e.g., contact interactions) or material nonlinearities (e.g., plasticity, nonlinear piezoelectricity, hyper-elasticity).

Nonlinear systems pose many practical challenges for simulation or reduction, which are highlighted as follows:

1. Nonlinear systems usually lack a closed-form solution. This implies that the input-output relationship cannot be analytically expressed via transfer functions or via convolution integrals. Moreover, nonlinear systems exhibit strong and complex behaviors such as bifurcation (e.g., jump phenomenon [214, 171]), several attractors (e.g., stable/unstable/limit-cycles), chaotic behavior, and multiple equilibrium points. This makes nonlinear systems harder to analyze and manipulate than linear systems.

2. Nonlinear dynamical systems also lack a canonical form structure of the reduced model. As a result, an efficient representation of the nonlinear system or its reduced model becomes highly inefficient. This is in contrast with linear time-invariant (LTI) systems which can be represented by several matrices $\mathbf{A}, \mathbf{B}, \mathbf{C},$ and $\mathbf{D}$.

3. There is a limited guarantee of the behavior of the reduced nonlinear model, i.e., the reduced model is expressed locally or for a given training data.

4. Reduction of nonlinear dynamical systems results in the reduction of only the state-vector dynamics. The nonlinear function still remains in the original high-dimensional space. This requires different hyper-reduction techniques "on-top-of " dimension reduction methods to effectively approximate the nonlinear inner-products.

5. Due to the lack of practical system-theoretic measures and analytical solutions, nonlinear systems are typically reduced in a "simulation-based" framework. Conventional nonlinear MOR methods such as proper orthogonal decomposition (POD) [306], trajectory piece-wise linear method (TPWL) [249, 250], empirical grammians [178, 179] or the reduced-basis method [236, 251] rely on taking expensive temporal measurements of the system to obtain the orthonormal basis for the reduced system. This imposes an enormous computational burden, especially when several simulation runs are required for different system parameters or training inputs.

1.3 Thesis objectives

This thesis focuses on "simulation-free" or system-theoretic methods for the reduction of large-scale nonlinear dynamical systems. MOR for linear systems is well explored, and several decent algorithms are available at our disposal. Thus, the goal of this thesis will be to explore the possibility of extending the already established MOR methods for linear systems to the nonlinear systems, and also develop novel frameworks that are better suited for a general class of nonlinear systems. The main objectives of the thesis are enlisted as follows:

1. The first and foremost goal will be to get insight into some famous nonlinear MOR (NMOR) methods like POD, TPWL, moment-matching, etc. With this, we will present some of the shortcomings of these methods, and based upon that, propose some novel NMOR frameworks that improve/enhance the current state-of-the-art methods. The central aim will be to enhance the simulation speeds or the large CPU core-hours associated while reducing large-scale nonlinear systems. For this, we will focus on using system-theoretic measures to improve both the offline and online CPU times involved. We will demonstrate the proposed frameworks on several nonlinear benchmark systems.

2. Another aim would be to reduce nonlinear systems in the second-order form. In this regard, we will focus on developing efficient reduction schemes that would preserve the structure of the original model. The reduced model, thus obtained, will be of the same form as that the original model to maintain the physical interpretation of the state variables. We will demonstrate the proposed methods on large-scale power system models. Unlike the common practice of linearizing the external area of the power system, we will focus on using nonlinear methods directly to effectively capture the nonlinear swing dynamics of power systems models.

3. Another central subject of concern in this thesis is the reduction of standard nonlinear systems having parametric dependencies. Since the traditional NMOR methods are not robust to parametric variation, thus these techniques cannot be applied directly to such models. As such, we will focus on developing novel parametric MOR (PMOR) methods in this direction.

4. The final objective of the thesis will be to develop efficient parametric sampling strategies for PMOR. This would involve developing system-theoretic methods in which the information extracted from the governing model will guide the selection of sampling points in the parametric domains. The result of adaptive sampling will enhance the approximation qualities of the reduced models while avoiding oversampling scenarios.

1.4 Scientific contributions

The main contributions of this thesis are highlighted as follows:

1. An accurate and low-cost projection framework is developed in Chapter 4 called the nonlinear moment-matching with discrete empirical interpolation method (NLMM-DEIM). The framework is used to obtain reduced models by solving the underlying Sylvester PDE after approximating it to nonlinear algebraic system using some numerical simplifications. The expensive cost of evaluating the nonlinear terms is reduced by using DEIM as a hyper-reduction method. The proposed scheme is implemented on several nonlinear benchmark models.

2. Again in Chapter 4, another reduction framework called NLMM-DMD is presented, by using NLMM with dynamic mode decomposition (DMD). The algorithm generates compact ROMs in a non-intrusive fashion. This is achieved by approximating the nonlinear function using the infinite dimensional Koopman operator by generating low-dimensional spatio-temporal modes of evolution.

3. In Chapter 5, the intrusive and non-intrusive MOR scheme based on NLMM-DMD is used to obtain a simulation-free reduction for power system models. Unlike the conventional linearization of external area of a power system model, we consider the nonlinear effective network (EN) and the synchronous motor (SM) power grid models. The results are substantiated using the reduced EN and SM models of the IEEE 118 and IEEE 300 bus systems for realistic fault situations. The numerical simulations are implemented in the MATLAB environment and the source codes are provided.

4. In the context of nonlinear models in second-order form, a numerical algorithm is presented in Chapter 5. The algorithm is called second-order moment-matching with discrete empirical interpolation method (SO-NLMM-DEIM) and is able to construct ROMs that preserve the second-order structure of the problem. The MATLAB implementation of SO-NLMM-DEIM is provided for reusability.

5. In the context of parametric nonlinear MOR, a numerical scheme is devised that constructs parametric ROMs by interpolating the neighboring reduced models obtained at regular intervals in the parameter space. In the scheme, we employ NLMM-DEIM to generate a pool of reduced coordinates and then transform each of them on a universal subspace for a meaningful interpolation. The numerical framework is demonstrated on a suite of nonlinear parametric systems in Chapter 6.

6. The matrix interpolatory framework for nonlinear systems is further modified in Chapter 7, and an adaptive sampling scheme is devised that uses

the information from principal angles of reduced subspaces to find optimal number of parametric samples for interpolation. The upfront offline cost of generating the training data is then amortized by the savings achieved by interpolation in online stage.

1.5 Outline of thesis

This thesis is divided into four parts. Part I discusses the preliminaries for the thesis and covers Chapters 1, and 2. After the introduction in Chapter 1, we begin with some mathematical preliminaries in Chapter 2 needed for the rest of the chapters. Therein, we discuss the projection-based technique to obtain ROM architecture in linear systems. We also present an overview of various linear MOR schemes and highlight the similarities and differences among various methods. Then, we describe the reduction via moment-matching in both time and frequency domain settings. Towards the end, we discuss some open issues and recent advancements in linear moment-matching.

Part II is dedicated to nonlinear state-space systems and covers Chapters 3, 4, and 5. In Chapter 3, we begin with the projection-based reduction of a general class of nonlinear systems. Therein, we discuss the linear and nonlinear Galerkin projection techniques to obtain nonlinear reduced models. We also provide a review of some of the famous nonlinear MOR methods. Then, we lay focus on the notion of nonlinear moment-matching via steady-state response matching. We describe the general methodology via theorems and diagrams. Finally, we discuss the hyper-reduction in nonlinear systems along-with some open issues in nonlinear MOR. In Chapter 4, we discuss the approximated NLMM by describing certain numerical simplification of the Sylvester PDE to arrive at a practical algorithm used to extract the basis for reduced subspace. Then, we propose two efficient reduction frameworks based on NLMM: (i) NLMM-DEIM, and (ii) NLMM-DMD. The former is inspired by POD-DEIM and avoids the expensive measurements needed in POD, while the latter is used to replace DEIM in a non-intrusive fashion. Both the schemes are validated and tested on several benchmark nonlinear systems and compared with POD and DEIM for reference. In Chapter 5, we discuss nonlinear state-systems models in second-order form. Therein, we first revisit the reduction of linear second-order models based on second-order Krylov methods. Then, we propose another reduction framework, i.e., SO-NLMM-DEIM, which is used to preserve the second-order structure in the reduced models. The scheme is then tested on a variety of second-order nonlinear models emerging from power systems. The results are compared with POD for reference.

Part III is devoted to reduction of nonlinear parametric state-space models and covers Chapters 6 and 7. We begin with the problem of parametric MOR for nonlinear systems in Chapter 6. Therein, we present a review of parametric MOR schemes, and afterward, we propose a nonlinear pMOR reduction framework using the matrix interpolatory framework, which is an extension of the matrix interpo-

latory method for linear systems. We then test the scheme on several parametric nonlinear models. Towards the end, we discuss some aspects of the scheme and mention some limitations for future research. In Chapter 7, we discuss one of the most concerning aspects in pMOR. i.e., the adaptive sampling of parametric space. We start with a general introduction of the topic and provide a review of available techniques. Then, we describe how subspace angles can be used to extract meaningful information that helps to resample the parametric domains for better quality ROMs. The methodology is presented for a two-dimensional parametric system, and a numerical algorithm is described that performs the adaptive search. The algorithm is tested on a suite of parametric nonlinear models emerging from circuit theory and MEMS.

The final Part IV completes the thesis with Chapter 8 describing the summary of work carried and highlighting the conclusions from each chapter along with an outlook for some future perspectives in the area of nonlinear model order reduction.

Chapter 2

Fundamentals & Review of Linear Model Order Reduction

2.1 Mathematical preliminaries

In the following, we describe some important mathematical fundamentals required for the rest of the thesis.

2.1.1 Projection

Most of the MOR schemes are employed via *projection* i.e., the dynamics of the FOM is projected onto a low-dimensional manifold. This enforces the higher-order dynamics of the FOM to evolve in a subspace which is much lower than the original model. In the following, we discuss some important key observations regarding projection from Refs.[204, 254, 152].

Definition 2.1 (Projector) A matrix $\mathbf{P} \in \mathbb{R}^{n \times n}$ is called a *projector* if the linear mapping from $\mathbb{R}^n$ to itself is idempotent [1] ▲

Using this definition, a few properties follow which are:

(a) If $\mathbf{P}$ is a projector onto $\mathcal{V}$, then $\mathbf{P}$ is the identity operator on $\mathcal{V}$, i.e., $\mathbf{Px} = \mathbf{x}, \forall\ \mathbf{x} \in \mathcal{V}$.

(b) If $\text{ran}(\mathbf{P}) = \mathcal{V}$, then $\mathbf{P}$ is said to the projector onto the subspace $\text{ran}(\mathbf{V}) = \mathcal{V}$.

(c) The complementary projector of $\mathbf{P}$ is given as $\mathbf{P}_\perp = \mathbf{I} - \mathbf{P}$.

(d) $\ker(\mathbf{P})$=$\text{ran}(\mathbf{I} - \mathbf{P}) = \mathcal{W}^\perp$ is called the *orthogonal complement* of $\text{ran}(\mathbf{W}) = \mathcal{W}$.

[1] A matrix $\mathbf{P}$ is idempotent if $\mathbf{P^2} = \mathbf{P}$

In order to project $\mathbf{x} \in \mathbb{R}^n$ onto the r-dimensional subspace $\mathcal{V}$ spanned by the columns of $\mathbf{V} \in \mathbb{R}^{n \times r}$, the projection is performed along the vector $\boldsymbol{\epsilon}$ which is orthogonal to subspace $\mathcal{W}$ spanned by columns of matrix $\mathbf{W} \in \mathbb{R}^{n \times r}$ (cf. Fig. 2.1). The projected vector $\mathbf{x}_{\text{proj}} \in \mathcal{V}$ is given by the linear combination of the basis vectors $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_r$, i.e.,

$$\mathbf{x}_{\text{proj}} = \sum_{i=1}^{r} \mathbf{v}_i \mathbf{c}_i = \mathbf{V}\mathbf{c}, \tag{2.1}$$

where $\mathbf{c}$ is the vector of unknown coefficients. From Fig. 2.1, we observe that

$$\mathbf{x} = \mathbf{x}_{\text{proj}} + \boldsymbol{\epsilon} \implies \boldsymbol{\epsilon} = \mathbf{x} - \mathbf{x}_{\text{proj}}, \tag{2.2}$$

and the columns of $\mathbf{W}$ are orthogonal to $\boldsymbol{\epsilon}$, i.e., $\mathbf{W}^T\boldsymbol{\epsilon} = 0$, as such, we have:

$$\mathbf{W}^T\boldsymbol{\epsilon} = \mathbf{W}^T\mathbf{x} - \mathbf{W}^T\mathbf{x}_{\text{proj}} = \mathbf{W}^T\mathbf{x} - \mathbf{W}^T\mathbf{V}\mathbf{c} = \mathbf{0}, \tag{2.3}$$

which when solved for $\mathbf{c}$ gives:

$$\mathbf{c} = (\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T\mathbf{x}. \tag{2.4}$$

Finally, substituting (2.4) in (2.1) yields the projection $\mathbf{x}_{\text{proj}}$ given as:

$$\mathbf{x}_{\text{proj}} = \mathbf{V}\mathbf{c} = \underbrace{\mathbf{V}(\mathbf{W}^T\mathbf{V})^{-1}\mathbf{W}^T}_{\mathbf{P}}\mathbf{x} = \mathbf{P}\mathbf{x}. \tag{2.5}$$

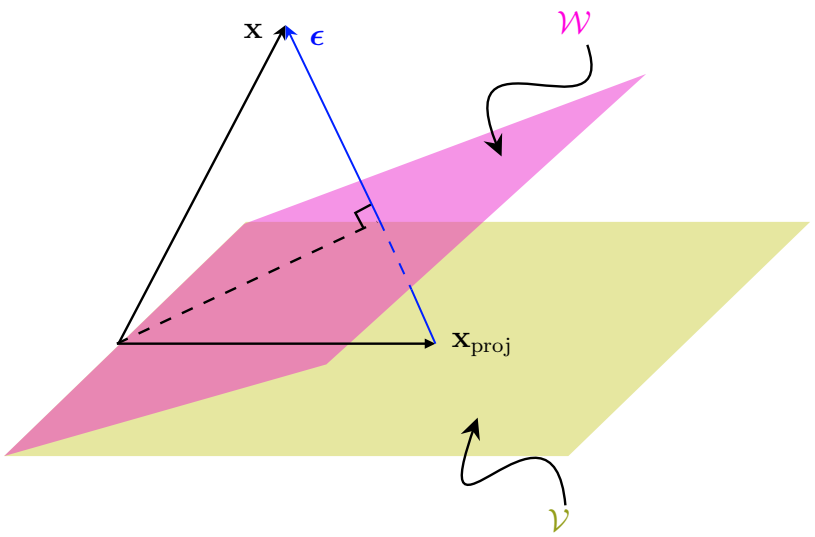

Figure 2.1: Projection of $\mathbf{x}$ onto $\mathbf{x}_{\text{proj}} \in \mathcal{V}$ along $\boldsymbol{\epsilon} \in \mathcal{W}^{\perp}$

Definition 2.2 (Orthogonal Projector) A projection $\mathbf{P}$ is called *orthogonal*, if null($\mathbf{P}$)=ran($\mathbf{P}$)$^{\perp}$ otherwise *oblique*. Thus, an orthogonal projector is defined thorough the following requirements:

$$\mathbf{P}\mathbf{x} \in \mathcal{V} \quad \text{and} \quad (\mathbf{I}-\mathbf{P})\mathbf{x} \perp \mathcal{V}. \tag{2.6}$$

▲

For the case of an orthogonal projection, the subspace $\mathcal{W} = \mathcal{V}$ (cf. Fig 2.1).

Remark 2.1 (Orthogonal Projectors) A projector is orthogonal if and only if it is *Hermitian*, i.e., $\mathbf{P} = \mathbf{P}^{\mathbf{H}}$

The following are the properties of orthogonal projectors:

(a) $\left\|\mathbf{x}\right\|_2^2 = \left\|\mathbf{P}\mathbf{x}\right\|_2^2 + \left\|(\mathbf{I}-\mathbf{P})\mathbf{x}\right\|_2^2$

(b) $\left\|\mathbf{P}\mathbf{x}\right\|_2 \leq \left\|\mathbf{x}\right\|_2$

(c) $\left\|\mathbf{P}\right\|_2 = 1$

(d) Any orthogonal projector has only two eigenvalues: zero or one.

2.1.2 Singular Value Decomposition (SVD)

The singular value decomposition is one of the most critical matrix decomposition tools of the computational era, providing a foundation for nearly all of the reduction methods presented in this thesis. The SVD method is guaranteed to exist and can be used for various purposes. In the context of the MOR, SVD is used as an underlying algorithm of principal component analysis (PCA) to extract low-rank approximations from time-displaced snapshot data obtained from models. SVD will provide a foundation for many MOR frameworks presented in this thesis. Generally, we are interested in analyzing a large data set $\mathbf{X} \in \mathbb{C}^{n \times n_s}$ given as:

$$\mathbf{X} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_1 & \mathbf{x}_2 & \cdots & \mathbf{x}_{n_s} \\ | & | & & | \end{bmatrix}, \tag{2.7}$$

where the columns $x_i \in \mathbb{C}^n$ can be measurements from an experiment or simulations. The column vectors may also represent the state of a physical system evolving in time, e.g., fluid velocity, state of a weather simulation, a set of neural measurements, etc. The columns of $\mathbf{X}$ are often known as *snapshots*, and n_s is the number of snapshots. Typically, $n >> m$ which results in a tall-skinny matrix.

Definition 2.3 (Singular Value Decomposition) Given a matrix $\mathbf{X} \in \mathbb{C}^{n \times n_s}$, the SVD is defined as:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*, \tag{2.8}$$

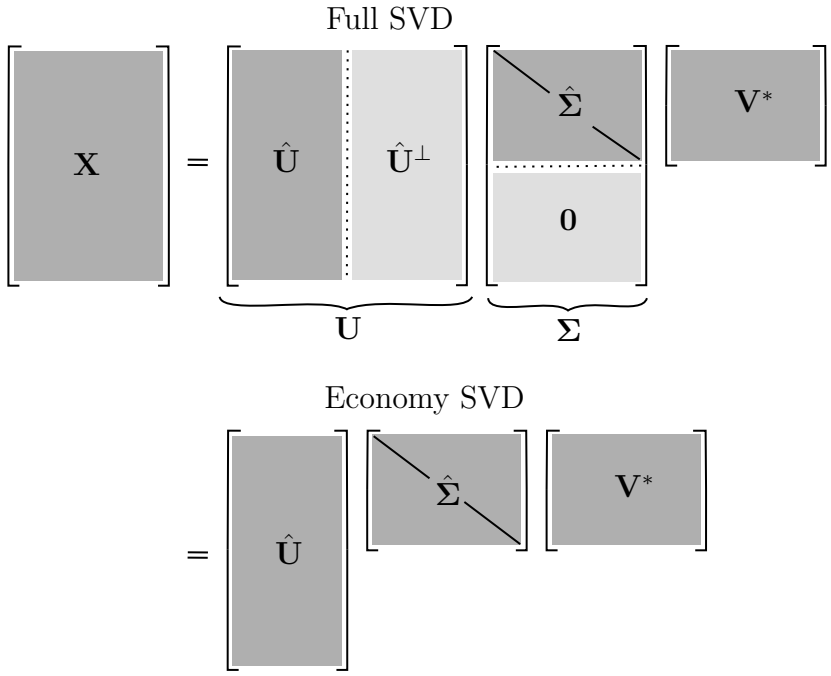

Figure 2.2: Schematic of matrices in the full and economy SVD [64]

where $\mathbf{\Sigma} \in \mathbb{C}^{n \times n_s}$ is a diagonal matrix with real, non-negative diagonal entries, and $\mathbf{U} \in \mathbb{C}^{n \times n}, \mathbf{V} \in \mathbb{C}^{n_s \times n_s}$ are unitary matrices [2] with orthonormal columns. ▲ When $n \geq n_s$, the matrix $\mathbf{\Sigma}$ has at most n_s non-zero entries on the diagonal, and is often written as $\mathbf{\Sigma} = \begin{bmatrix} \hat{\mathbf{\Sigma}} \\ \mathbf{0} \end{bmatrix}$. Thus, an *economy* version of the SVD can be obtained as follows:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^* = \begin{bmatrix} \hat{\mathbf{U}} & \hat{\mathbf{U}}^\perp \end{bmatrix} \begin{bmatrix} \hat{\mathbf{\Sigma}} \\ \mathbf{0} \end{bmatrix} \mathbf{V}^* = \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\mathbf{V}^*, \tag{2.9}$$

where $\hat{\mathbf{U}}, \hat{\mathbf{U}}^\perp$ have r, $n - r$ columns. The columns of $\hat{\mathbf{U}}^\perp$ span a vector space which is orthogonal and complementary to that spanned by $\hat{\mathbf{U}}$. Furthermore, the columns of $\mathbf{U}$ are the *left singular vectors* of $\mathbf{X}$, and the columns of $\mathbf{V}$ are the *right singular vectors*. The diagonal entries of $\hat{\mathbf{\Sigma}}$ are known as the *singular values* σ_i $\{i = 1, .., n\}$ of $\mathbf{X}$, and are ordered from largest to smallest. The schematic for full and economic version of the SVD method are presented in Fig. 2.2.

Remark 2.2 (SVD) These singular values are *eigenvectors* of $\mathbf{X}\mathbf{X}^*$ and $\mathbf{X}^*\mathbf{X}$ respectively. It readily follows that

$$\mathbf{X}\mathbf{v}_i = \sigma_i \mathbf{u}_i \qquad i = 1, ..., n. \tag{2.10}$$

[2] A square matrix $\mathbf{U}$ is unitary if $\mathbf{U}^*\mathbf{U} = \mathbf{I}$

The SVD has certain important properties, which are stated as follows.

(a) Rank $\mathbf{X} = r$.

(b) The orthogonal projection onto the span of columns of $\mathbf{X}$ is $\hat{\mathbf{U}}\hat{\mathbf{U}}^{\perp}$.

(c) The orthogonal projection onto the kernel of $\mathbf{X}^*$ is $\mathbf{I}_n - \hat{\mathbf{U}}\hat{\mathbf{U}}^*$.

(d) The Frobenius norm of $\mathbf{X}$ is $\left\|\mathbf{X}\right\|_{\mathrm{F}} = \sqrt{\sigma_1^2 + ... + \sigma_n^2}$.

Perhaps the most important property of SVD is that it gives an *optimal* low-rank approximation to a matrix $\mathbf{X}$ which is given by the Schmidt-Eckart-Young-Mirsky theorem as follows:

Theorem 2.1 (Schmidt-Eckart-Young-Mirsky) *[98] The optimal rank-r approximation to* $\mathbf{X}$*, in a least-square sense, is given:*

$$\underset{\tilde{\mathbf{X}},\ \text{s.t. rank}(\tilde{\mathbf{X}})=r}{\arg\min} \left\|\mathbf{X} - \tilde{\mathbf{X}}\right\|_{\mathrm{F}} = \tilde{\mathbf{U}}\tilde{\mathbf{\Sigma}}\tilde{\mathbf{V}}^*, \tag{2.11}$$

▲

where $\tilde{\mathbf{\Sigma}}$ contains the leading $r \times r$ sub-block of $\mathbf{\Sigma}$ and $\tilde{\mathbf{U}}, \tilde{\mathbf{V}}$ represent the first r leading columns of $\mathbf{U}$ and $\mathbf{V}$ respectively. Since $\mathbf{\Sigma}$ is diagonal, the rank r SVD approximation is written as the *dyadic* summation of r distinct rank-1 matrices as follows:

$$\tilde{\mathbf{X}} = \sum\nolimits_{i=1}^{r} \sigma_i \mathbf{u}_i \mathbf{v}_i^* = \sigma_1 \mathbf{u}_1 \mathbf{v}_1^* + \sigma_2 \mathbf{u}_2 \mathbf{v}_2^* + ... + \sigma_r \mathbf{u}_r \mathbf{v}_r^*. \tag{2.12}$$

Thus, the high-dimensional matrix $\mathbf{X}$ can be described by a few dominant patters given by the columns of $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{V}}$. This has the benefit of reducing the size and dimension of large data sets, yielding a tractable basis for visualization and analysis. The use of SVD in this thesis can found in Sec. 4.1.2, 4.3.1, 5.3, and Sec. 6.3.

2.1.3 Krylov subspaces

In this thesis, we are mainly concerned with employing projections onto so-called *Krylov subspaces*. These are named after the Russian mathematician A. N. Krylov, who introduced it in 1931 for the computation of characteristic polynomial of a matrix. Herein we describe the different types of Krylov subspaces:

A. Classical Krylov subspace:

Definition 2.4 (Classical Krylov subspace) Given a matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ and a starting vector $\mathbf{b} \in \mathbb{R}^n$, the r-order Krylov subspace, denoted as $\mathcal{K}_r(\mathbf{A}, \mathbf{b})$, is defined as:

$$\mathcal{K}_r(\mathbf{A}, \mathbf{b}) = \mathrm{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, ..., \mathbf{A}^{r-1}\mathbf{b}\}. \tag{2.13}$$

If the r vectors are linearly independent, then they form a r-dimensional basis $\mathcal{V} = \text{ran}(\mathbf{V}) \subset \mathcal{K}_r(\mathbf{A}, \mathbf{b})$ with the basis matrix as $\mathbf{V} \in \mathbb{R}^{n \times r}$. In control system community, the Krylov subspaces are known as *reachability* or *controllability subspaces*. ▲

In practice, the powers of $\mathbf{A}$ are computed iteratively from the previous ones by setting $\mathbf{v}_0 = \mathbf{b}, \mathbf{v}_l = \mathbf{A}\mathbf{v}_{l-1}, l = 1, 2, ...r-1$, with a subsequent orthogonalization at every iteration using a Gram-Schmidt process. The most common implementations are the *Lanczos*[180] method and the *Arnoldi* iteration [22].

B. Rational Krylov subspace:
Proposed by Ruhe [253], the rational Krylov subspaces are used to obtain *shifted* versions of the classical Krylov subspaces.

Definition 2.5 (Block rational Krylov subspaces) Given the matrices $\mathbf{E}, \mathbf{A} \in \mathbb{R}^{n \times n}, \mathbf{B} \in \mathbb{R}^{n \times m}$, and the interpolation point $\sigma \in \mathbb{C}$ ($\sigma \notin \lambda(\mathbf{E^{-1}A})$), the r-order *block* rational Krylov subspace $\mathcal{K}_r(\mathbf{A}_\sigma^{-1}\mathbf{E}, \mathbf{A}_\sigma^{-1}\mathbf{B})$ is defined as:

$$\mathcal{K}_r(\mathbf{A}_\sigma^{-1}\mathbf{E}, \mathbf{A}_\sigma^{-1}\mathbf{B}) = \text{span}\{\mathbf{A}_\sigma^{-1}\mathbf{B}, \mathbf{A}_\sigma^{-1}\mathbf{E}\mathbf{A}_\sigma^{-1}\mathbf{B}, ..., (\mathbf{A}_\sigma^{-1}\mathbf{E})^{r-1}\mathbf{A}_\sigma^{-1}\mathbf{B}\}, \quad (2.14)$$

where $\mathbf{A}_\sigma = (\sigma\mathbf{E} - \mathbf{A})^{-1}$. ▲

A key advantage of Krylov subspaces is that an orthonormal basis $\mathbf{V}$ can be constructed that will span the union of several Krylov subspaces. For instance, the following projection basis $\mathbf{V}$, with:

$$\text{Ran}(\mathbf{V}) \subseteq \text{span}\{\mathbf{E}^{-1}\mathbf{B}, ..., (\mathbf{E}^{-1}\mathbf{A})^{r-1}\mathbf{E}^{-1}\mathbf{B}, \mathbf{A}^{-1}\mathbf{B}, ..., (\mathbf{A}^{-1}\mathbf{E})^{r-1}\mathbf{A}^{-1}\mathbf{B}\}, \quad (2.15)$$

is obtained as the the union of rational Krylov subspaces $\mathcal{K}_r(\mathbf{E}^{-1}\mathbf{A}, \mathbf{E}^{-1}\mathbf{B}) \cup \mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{B})$ for $\sigma = \infty$ and $\sigma = 0$. This extended version of the Krylov subspace is referred as *Krylov-Plus-Inverted-Krylov (KPIK)* .

Another possibility is to obtain different shifts $\sigma_i, i = 1, ..., q$ with respective multiplicities r_i. As such, the union of Krylov subspaces $\mathcal{K}_{r_1}(\mathbf{A}_{\sigma_1}^{-1}\mathbf{E}, \mathbf{A}_{\sigma_1}^{-1}\mathbf{B}) \cup ... \cup \mathcal{K}_{r_q}(\mathbf{A}_{\sigma_q}^{-1}\mathbf{E}, \mathbf{A}_{\sigma_q}^{-1}\mathbf{B})$ leads to the block *multipoint* rational Krylov subspace given as:

$$\text{Ran}(\mathbf{V}) \subseteq \text{span}\{\mathbf{A}_{\sigma_1}^{-1}\mathbf{B}, ..., (\mathbf{A}_{\sigma_1}^{-1}\mathbf{E})^{r_1-1}\mathbf{A}_{\sigma_1}^{-1}\mathbf{B}, ..., \mathbf{A}_{\sigma_q}^{-1}\mathbf{B}, ..., (\mathbf{A}_{\sigma_q}^{-1}\mathbf{E})^{r_q-1}\mathbf{A}_{\sigma_q}^{-1}\mathbf{B}\}, \quad (2.16)$$

where the matrix $\mathbf{A}_{\sigma_i} = (\sigma_i\mathbf{E} - \mathbf{A})$.

C. Tangential Krylov subspace:
A drawback of the block Krylov subspaces is that at every new step, the basis grows by m. In order to suppress the fast growth of dimension, one can choose a single tangential direction $\mathbf{w} \in \mathbb{C}^m$ and then apply the vector $\mathbf{Bw}$. The result will be that the number of new columns per iteration will be reduced to one. Furthermore, it is also possible to select one tangential direction $\{\mathbf{w}_i\}_{i=1}^q$ for each shift $\{\sigma_i\}_{i=1}^q$ leading to *tangential multipoint* Krylov subspace given as:

$$\text{Ran}(\mathbf{V}) \subseteq \text{span}\{\mathbf{A}_{\sigma_1}^{-1}\mathbf{Bw}_1, \mathbf{A}_{\sigma_2}^{-1}\mathbf{Bw}_2, ..., \mathbf{A}_{\sigma_q}^{-1}\mathbf{Bw}_q\}. \tag{2.17}$$

The tangential Krylov subspaces constitute a very general representation, since the choice of $\mathbf{w}_i = \mathbf{I}_m$ leads to a block Krylov subspace and for $m = 1$ simplifies to $\mathbf{B} \to \mathbf{b}$ and $\mathbf{w}_i = 1$.

Remark 2.3 (Output Krylov subspaces) By replacing $\mathbf{A} \to \mathbf{A}^T, \mathbf{E} \to \mathbf{E}^T$ and $\mathbf{B} \to \mathbf{C}^T$, the dual counterparts for all stated Krylov subspaces can be obtained. These are also known as *output Krylov subspaces* ▲

Krylov subspaces are extensively used in the linear algebra community for:

(a) *Iterative solution of linear system* $\mathbf{Ax} = \mathbf{b}$: Based on the fact that the successive approximations of $\mathbf{x}$ belong to the Krylov subspace $\mathcal{K}()$, both Arnoldi and Lanczos algorithms are used to iteratively construct orthonormal basis of these subspaces.

(b) *Iterative computation of eigenvalues of* $\mathbf{A}$: The simplest approach is by using the *power method* whereby, the successive terms $\mathbf{A}^{k-1}\mathbf{b}$ are computed for a given $\mathbf{b}$. However, due to slow convergence, Krylov subspace methods are used where at the kth step, the information contained in $\mathbf{b}, \mathbf{Ab}, ..., \mathbf{A}^{k-1}\mathbf{b}$ is used [184, 275].

(c) *Approximation of linear systems by moment-matching*: This is the problem of interest in this thesis and is discussed in detail.

2.1.4 Error measures

Throughout this thesis, we will use certain error margins to access the quantitative comparison of the reduced models with the original model. As such, we describe herein the quantitative analysis of ROMs in the time-domain, which is mainly employed in the context of nonlinear dynamical systems. For linear systems, the error is usually measured in frequency-domain using transfer functions of FOMs and ROMs. Based on the specific application, the accuracy of ROMs can be measured in terms of whole state vector $\mathbf{x}(t)$ or through some specific measurement vector $\mathbf{y}(t)$.

A. Norm-wise error measure

For a norm-wiser error evaluation, the simulated data is collected and evaluated as matrices, i.e. $\mathbf{x} \in \mathbb{R}^{n\times n_s}, \mathbf{x}_r \in \mathbb{R}^{r\times n_s}$ and $\mathbf{y}, \mathbf{y}_r \in \mathbb{R}^{p\times n_s}$. Now, depending upon the type of norm used, the relative errors for various reduction methods "*red*" can be obtained *norm-wise* as follows:

$$\mathbf{e}^{red}_{\mathbf{x},\mathrm{rel}(*)} = \frac{\left\|\mathbf{x} - \mathbf{V}\mathbf{x}^{red}_r\right\|_{(*)}}{\left\|\mathbf{x}\right\|_{(*)}}, \qquad \mathbf{e}^{red}_{\mathbf{y},\mathrm{rel}(*)} = \frac{\left\|\mathbf{y} - \mathbf{y}^{red}_r\right\|_{(*)}}{\left\|\mathbf{y}\right\|_{(*)}}, \tag{2.18}$$

where any desired *matrix* norm $(*) \in \{1, 2, \infty, \mathrm{F}\}$ can be used. Additionally, the norm-wise error can also be measured in time using a desired *signal* norm $\mathcal{L}_p \in \{\mathcal{L}_1, \mathcal{L}_2, \mathcal{L}_\infty\}$ as follows:

$$\mathbf{e}^{red}_{\mathbf{x},\mathrm{rel},\mathcal{L}_\mathbf{p}} = \frac{\left\|\mathbf{x} - \mathbf{V}\mathbf{x}^{red}_r\right\|_{\mathcal{L}_p}}{\left\|\mathbf{x}\right\|_{\mathcal{L}_p}}, \qquad \mathbf{e}^{red}_{\mathbf{y},\mathrm{rel},\mathcal{L}_\mathbf{p}} = \frac{\left\|\mathbf{y} - \mathbf{y}^{red}_r\right\|_{\mathcal{L}_p}}{\left\|\mathbf{y}\right\|_{\mathcal{L}_p}}, \tag{2.19}$$

where

$$\mathbf{e}^{red}_{\mathbf{y},\mathrm{rel},\mathcal{L}_\mathbf{1}} = \frac{\left\|\mathbf{y} - \mathbf{y}^{red}_r\right\|_{\mathcal{L}_1}}{\left\|\mathbf{y}\right\|_{\mathcal{L}_1}} = \frac{\sum_{k=1}^{n_s}\left\|\mathbf{y}(t_k) - \mathbf{y}^{red}_r(t_k)\right\|_2}{\sum_{k=1}^{n_s}\left\|\mathbf{y}(t_k)\right\|_2}, \tag{2.20a}$$

$$\mathbf{e}^{red}_{\mathbf{y},\mathrm{rel},\mathcal{L}_\mathbf{2}} = \frac{\left\|\mathbf{y} - \mathbf{y}^{red}_r\right\|_{\mathcal{L}_2}}{\left\|\mathbf{y}\right\|_{\mathcal{L}_2}} = \frac{\sqrt{\sum_{k=1}^{n_s}\left\|\mathbf{y}(t_k) - \mathbf{y}^{red}_r(t_k)\right\|_2^2}}{\sqrt{\sum_{k=1}^{n_s}\left\|\mathbf{y}(t_k)\right\|_2^2}}, \tag{2.20b}$$

$$\mathbf{e}^{red}_{\mathbf{y},\mathrm{rel},\mathcal{L}_\infty} = \frac{\left\|\mathbf{y} - \mathbf{y}^{red}_r\right\|_{\mathcal{L}_\infty}}{\left\|\mathbf{y}\right\|_{\mathcal{L}_\infty}} = \frac{\max_k\left\|\mathbf{y}(t_k) - \mathbf{y}^{red}_r(t_k)\right\|_2}{\max_k\left\|\mathbf{y}(t_k)\right\|_2}. \tag{2.20c}$$

B. Point-wise error measure:

Besides the norm-wise error, the error can also be measured as a time-series data by evaluating the states $\mathbf{x}(t_k) \in \mathbb{R}^n, \mathbf{x}^{red}_r(t_k) \in \mathbb{R}^r$ and the outputs $\mathbf{y}(t_k), \mathbf{y}^{red}_r(t_k) \in \mathbb{R}^p$ at every time-step t_k. The *point-wise* wise in this case will be evaluated as follows:

$$\mathbf{e}^{red}_{\mathbf{x},\mathrm{rel}(.)}(t_k) = \frac{\left\|\mathbf{x}(t_k) - \mathbf{V}\mathbf{x}^{red}_r(t_k)\right\|_{(.)}}{\left\|\mathbf{x}(t_k)\right\|_{(.)}}, \qquad \mathbf{e}^{red}_{\mathbf{y},\mathrm{rel}(.)}(t_k) = \frac{\left\|\mathbf{y}(t_k) - \mathbf{y}^{red}_r(t_k)\right\|_{(.)}}{\left\|\mathbf{y}(t_k)\right\|_{(.)}} \tag{2.21}$$

for any desired *vector* norm $(.) \in \{1, 2, \infty, ...\}$.

2.2 Model order reduction in linear systems

2.2.1 Linear time-invariant systems

Surprisingly, many technical systems can be represented as linear time invariant (LTI) systems. Though most of the complex systems are intrinsically nonlinear, linear models often help to provide a first approximation of the underlying dynamics. Furthermore, in many practical situations, linearizing the nonlinear model around a known equilibrium point provides a reflection of how the original model evolves which is highly beneficial in control and design.

Let us begin with the state-space representation of a large-scale, continuous-time, multiple-input, multiple-output LTI system as follows:

$$\boldsymbol{\Sigma}\begin{cases}\mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), & \mathbf{x}(0) = \mathbf{x_0} \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}, \end{cases} \tag{2.22}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ is known as the *state-vector* in control system parlance. This represents the vector of unknowns whose entries are called *internal variables* or *state variables*. The order of the state-space model is n i.e., the state vector $\mathbf{x}(t) \in \mathbb{R}^n$ spans an n-dimensional Euclidean space. Vectors $\mathbf{u}(t) \in \mathbb{R}^m$ and $\mathbf{y}(t) \in \mathbb{R}^p$ are the inputs and outputs of the system, whereas $\mathbf{E} \in \mathbb{R}^{n\times n}$ is the (non-singular) descriptor matrix, $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the system matrix, $\mathbf{B} \in \mathbb{R}^{n\times m}$ is the input matrix, and $\mathbf{C} \in \mathbb{R}^{p\times n}$ is the output matrix respectively. System (2.22) represents a multiple-input multiple-output (MIMO) system with $m, p << n$. However, for single-input single-output (SISO) systems, $m = p = 1$, and the matrices $\mathbf{B}$, $\mathbf{C}$ become vectors $\mathbf{b}$ and $\mathbf{c}^T$ whereas vectors $\mathbf{u}$ and $\mathbf{y}$ become scalars u and y respectively. The LTI system (2.22) is called *regular* if the matrix $\mathbf{E}$ is nonsingular or *descriptor system* otherwise. In case of singular $\mathbf{E}$, (2.22) represents a system of differential-algebraic equations (DAEs) rather than ordinary differential equations (ODEs) which are more difficult to solve. So, system (2.22) essentially represents a large-scale, initial value problem that we are interested in solving for some predefined time-span, in the presence of some excitation function $\mathbf{u}(t)$.

Throughout this thesis the matrix $\mathbf{E}$ is assumed to be regular, i.e, without any algebraic constraints. The regularity of $\mathbf{E}$ theoretically allows us to replace $\mathbf{A} \to \mathbf{E}^{-1}\mathbf{A}$ and $\mathbf{B} \to \mathbf{E}^{-1}\mathbf{B}$ in order to obtain an *explicit* representation with $\mathbf{E} = \mathbf{I}$. Nevertheless, the inverse is usually avoided in practice due to numerical reasons and only used in theoretical statements. In addition to this, it is assumed throughout thesis that system 2.22 is *minimal*, i.e., the pair $(\mathbf{E}^{-1}\mathbf{A}, \mathbf{E}^{-1}\mathbf{B})$ is controllable (cf. Def. 2.7) and the pair $(\mathbf{C}, \mathbf{E}^{-1}\mathbf{A})$ is observable (cf. Def. 2.8). It is further assumed that the system $\boldsymbol{\Sigma}$ is asymptotically stable (cf. Def. 2.6) implying that all the eigenvalues are in the open left-half of the complex plane, i.e., $\lambda(\mathbf{E}^{-1}\mathbf{A}) \subset \mathbb{C}_-$.

Definition 2.6 (Asymptotic Stability) A dynamical system is called *stable*, if for every state-trajectory $\mathbf{x}(t)$ it holds $\left\|\mathbf{x}(t)\right\| \leq c, \forall t$ for some constant c and *asymptotically* stable if additionally, $\lim_{t\to\infty}\left\|\mathbf{x}(t)\right\| = 0$ for an arbitrary norm $\left\|.\right\|$ ▲

Definition 2.7 (Reachability) Given a dynamical system $\mathbf{\Sigma}$, a state $\hat{\mathbf{x}} \in \mathbb{R}^n$ is called *reachable* from the zero state if there exists a forcing function $\mathbf{u}(t)$ of finite energy, and a time $T < \infty$ such that

$$\hat{\mathbf{x}} = \phi(\mathbf{u}; 0; T).$$

The system $\mathbf{\Sigma}$ is called *reachable* if the reachability matrix given as

$$\mathcal{R} = \begin{bmatrix}\mathbf{B} & \mathbf{AB} & \cdots & \mathbf{A}^{n-1}\mathbf{B}\end{bmatrix} \in \mathbb{R}^{n\times nm}, \tag{2.23}$$

has full row rank, i.e., $\text{rank}(\mathcal{R}) = n$.
Alternatively, the linear system $\mathbf{\Sigma}$ is reachable if the *reachability gramian* $\mathcal{P} \in \mathbb{R}^{n\times n}$, defined as

$$\mathcal{P} = \int_{\tau=0}^{\infty} e^{(\mathbf{E}^{-1}\mathbf{A})\tau}\mathbf{E}^{-1}\mathbf{B}(\mathbf{E}^{-1}\mathbf{B})^T e^{(\mathbf{E}^{-1}\mathbf{A})^T\tau}\, d\tau, \tag{2.24}$$

is positive definite, i.e., $\mathcal{P} = \mathcal{P}^T \succ \mathbf{0}$. ▲

The concept of reachability is essential for some model reduction methods, such as balanced truncation, which will be discussed later on. The idea is to figure out which states are challenging to reach, i.e., states which require a large amount of energy to be reached. Once identified, those states can be neglected without influencing the behavior of the system.

Definition 2.8 (Observability) Given a dynamical system $\mathbf{\Sigma}$, the pair $(\mathbf{C}, \mathbf{A})$ is called *observable*, if the observability matrix defined as:

$$\mathcal{O} = \begin{bmatrix}\mathbf{C^T} & \mathbf{A^T C^T} & \cdots & \mathbf{A^T}^{n-1}\mathbf{C^T}\end{bmatrix}^T \in \mathbb{R}^{pn\times n}, \tag{2.25}$$

has full column rank, i.e., $\text{rank}(\mathcal{O}) = n$.
Alternatively, the linear system $\mathbf{\Sigma}$ is observable if and only if the *observability Gramian* $\mathcal{Q} \in \mathbb{R}^{n\times n}$, defined as

$$\mathcal{Q} = \int_{\tau=0}^{\infty} e^{(\mathbf{E}^{-1}\mathbf{A}^T)\tau}\mathbf{C}^T\mathbf{C}e^{(\mathbf{E}^{-1}\mathbf{A})\tau}\, d\tau, \tag{2.26}$$

is positive definite, i.e., $\mathcal{Q} = \mathcal{Q}^T \succ \mathbf{0}$. ▲

2.2.2 Input-output characterization in time domain

It is well known that the solution of the state-equation (2.22) is determined as:

$$\mathbf{x}(t) = e^{(\mathbf{E}^{-1}\mathbf{A})t}\mathbf{x_0} + \int_0^t e^{\mathbf{E}^{-1}\mathbf{A}(t-\tau)}\mathbf{E}^{-1}\mathbf{B}\mathbf{u}(\tau)\ d\tau. \tag{2.27}$$

Assuming a zero initial condition $\mathbf{x_0} = \mathbf{0}$, the so-called *impulse* response matrix of the system is given as:

$$\mathbf{h}(t) = \mathbf{C}e^{(\mathbf{E}^{-1}\mathbf{A})t}\mathbf{E}^{-1}\mathbf{B} \in \mathbb{R}^{p\times m}, \tag{2.28}$$

and the output response $\mathbf{y}_i(t)$ for $i = 1, 2, ..., p$ is then given as:

$$\mathbf{y}_i(t) = \sum_{k=1}^{m}\int_{\tau=0}^{t} h_{ik}(t-\tau)\mathbf{u}_k d\tau + \mathbf{c}_i^T e^{(\mathbf{E}^{-1}\mathbf{A})t}\mathbf{x_0}, \tag{2.29}$$

where denotes the (ik)-th entry of the impulse response matrix $\mathbf{h}(t)$.

2.2.3 Input-output characterization in frequency-domain

In order to derive an explicit input-output relation of system (2.22) in frequency-domain, we proceed by taking *Laplace transform* of equations (2.22)

$$s\mathbf{E}\mathbf{X}(s) - \mathbf{X}(0) = \mathbf{A}\mathbf{X}(s) + \mathbf{B}\mathbf{U}(s), \tag{2.30a}$$

$$\mathbf{Y}(s) = \mathbf{C}\mathbf{X}(s). \tag{2.30b}$$

Solving for state equation and substituting the result in output equation yields:

$$\mathbf{Y}(s) = \underbrace{\mathbf{C}(s\mathbf{E}-\mathbf{A})^{-1}\mathbf{B}}_{\mathbf{G}(s)}\mathbf{U}(s) + \mathbf{C}(s\mathbf{E}-\mathbf{A})^{-1}\mathbf{X}(0), \tag{2.31}$$

where $\mathbf{G}(s) = \mathbf{C}(s\mathbf{E}-\mathbf{A})^{-1}\mathbf{B} \in \mathbb{R}^{p\times m}$ is the rational transfer function matrix. For SISO systems, the transfer function is a rational function of degree n. A useful representation of the transfer function $\mathbf{G}(s)$ known as the *pole-residue expression* is obtained for a diagonal $\mathbf{A}$, which given as follows:

$$\mathbf{G}(s) = \sum_{i=1}^{n}\frac{R_i}{s-\lambda_i}, \tag{2.32}$$

where

$$R_i = \lim_{s\to\lambda_i}\mathbf{G}(s)(s-\lambda_i), \tag{2.33}$$

and $\lambda_i, i = 1, ..., n$ represent the eigenvalues of $\mathbf{A}$. Kindly note that *minimal* assumption of system $\mathbf{\Sigma}$ implies that no pole-zero cancellation occurs. This will otherwise result in a system with less dimensions.

2.2.4 Reduction using projection

As previously discussed in Section 2.1.1, model order reduction of dynamical systems is often performed using projection technique, i.e., the high-order dynamics are projected onto a subspace of lower dimension. In what follows, we explain the projective MOR scheme for LTI systems.

The idea is based on the premise that the state-trajectory $\mathbf{x}(t)$ mainly evolves in a r-dimensional subspace $\mathcal{V}$ of the state-space $\mathbb{R}^n$. Let $\mathbf{V} \in \mathbb{R}^{n\times r}$, which is full column rank matrix, be the basis of $\mathcal{V}$, then the state-vector $\mathbf{x}(t) \in \mathbb{R}^n$ can be expressed in terms of reduced state-vector $\mathbf{x}_r(t) \in \mathbb{R}^r$ as follows:

$$\mathbf{x}(t) = \mathbf{V}\mathbf{x}_r(t) + \mathbf{e}(t), \tag{2.34}$$

where $e(t)$ is the error due to approximation. Using (2.34) into the state-equation (2.22) yields an over-determined system of equations given as:

$$\mathbf{E}\mathbf{V}\dot{\mathbf{x}}_r(t) = \mathbf{A}\mathbf{V}\mathbf{X}_r(t) + \mathbf{B}\mathbf{u}(t) + \underbrace{\mathbf{A}\mathbf{e}(t) - \mathbf{E}\dot{\mathbf{e}}(t).}_{\mathbf{r}(t)} \tag{2.35}$$

Now, in order to make this a well-posed problem, the system (2.35) is projected orthogonally to another subspace $\mathcal{W} = \text{ran}(\mathbf{W})$ where the matrix $\mathbf{W} \in \mathbb{R}^{n\times r}$ is a basis of $\mathcal{W}$ and is selected such that $\mathbf{W}^T\mathbf{E}\mathbf{V}$ is non-singular. This yields the desired projector onto the subspace $\mathcal{U} = \text{ran}(\mathbf{E}\mathbf{V})$, and is given as:

$$\mathbf{P} = \mathbf{E}\mathbf{V}(\mathbf{W}^T\mathbf{E}\mathbf{V})^{-1}\mathbf{W}^T. \tag{2.36}$$

Multiplying (2.35) from the left with the projector (2.36) yields:

$$\mathbf{P}(\underbrace{\mathbf{E}\mathbf{V}\dot{\mathbf{x}}_r(t) - \mathbf{A}\mathbf{V}\mathbf{x}_r(t) - \mathbf{B}\mathbf{u}(t)}_{\mathbf{\Gamma}(\mathbf{V}\mathbf{x}_r(t)),\mathbf{u}(t)} - \mathbf{r}(t)) = \mathbf{0}. \tag{2.37}$$

Now, enforcing a so-called *Petro-Galerkin condition*, i.e., $\mathbf{W}^T\mathbf{r}(t) = \mathbf{0}$ eliminates the residual term leading to the final reduced model of dimension r as follows:

$$\mathbf{\Sigma}_r : \begin{cases} \mathbf{E}_r\dot{\mathbf{x}}_r(t) = \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{B}_r\mathbf{u}(t), & \mathbf{x}_r(0) = \mathbf{x}_{r,0}, \\ \mathbf{y}_r(t) = \mathbf{C}_r\mathbf{x}_r(t), \end{cases} \tag{2.38}$$

where

$$\mathbf{E}_r = \mathbf{W}^T\mathbf{E}\mathbf{V}, \mathbf{A}_r = \mathbf{W}^T\mathbf{A}\mathbf{V}, \mathbf{B}_r = \mathbf{W}^T\mathbf{B}, \mathbf{C}_r = \mathbf{C}\mathbf{V}, \tag{2.39}$$

and $\mathbf{x}_{r,0} = (\mathbf{W}^T\mathbf{E}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{E}\mathbf{x}_0$. It still remains open how to choose the projection matrices $\mathbf{V}$ and $\mathbf{W}$ such that $\mathbf{e}(t) = \left\|\mathbf{y} - \mathbf{y}_r\right\|$ is minimized. This is where the different reduction methods vary, and is explained next.

2.2.5 Overview of linear reduction methods

A. Methods based on balancing

Balanced truncation and the related methods are the methods of choice for model reduction techniques for LTI systems as it preserves the asymptotic stability in the reduced model. The idea was first introduced by Mullis and Roberts in 1976 [212], and was later formalized by Moore [210]. The idea is to transform the system into a realization where the states that are difficult to control are simultaneously hard to observe. This is achieved by simultaneously diagonalizing the reachability and observability gramians $\mathcal{P}$ and $\mathcal{Q}$ which are the solutions of the *Lyapunov* equations given as:

$$\mathbf{A}\mathcal{P}\mathbf{E}^T + \mathbf{E}\mathcal{P}\mathbf{A}^T + \mathbf{B}\mathbf{B}^T = \mathbf{0}, \tag{2.40a}$$

$$\mathbf{A}^T\mathbf{Q}\mathbf{E} + \mathbf{E}^T\mathbf{Q}\mathbf{A} + \mathbf{C}^T\mathbf{C} = \mathbf{0}, \tag{2.40b}$$

where $\mathbf{Q} = \mathbf{E}^{T^{-1}}\mathcal{Q}\mathbf{E}^{-1}$.

Definition 2.9 (Lyapunov balancing) The reachable, observable and stable system $\mathbf{\Sigma}$ is called *Lyapunov balanced* if

$$\mathcal{P} = \mathcal{Q} = \mathrm{diag}(\sigma_1, ..., \sigma_n), \tag{2.41}$$

where $\sigma_1 > \sigma_2 > ... > \sigma_n \geq 0$, and $\sigma_i = \sqrt{\lambda_i(\mathcal{P}\mathcal{Q})}$ are called *Hankel sinular values*. The systems with equal and diagonal Gramians are called *balanced realizations*. ▲

To be more precise, the smallest amount of energy required to reach a state $\mathbf{x}_r$ from zero initial state $\mathbf{x}_0$ is given by [14]

$$\mathbf{E}_r = \mathbf{x}_r^*\mathcal{P}^{-1}\mathbf{x}_r, \tag{2.42}$$

while the largest observation energy produced by the state $\mathbf{x}_0$ with no excitation function is given by (Chapter 4,[14])

$$\mathbf{E}_o = \mathbf{x}_0^*\mathcal{Q}\mathbf{x}_0. \tag{2.43}$$

Thus, if a system $\mathbf{\Sigma}$ satisfies the relation (2.41), then the states that require a huge amount of energy to reach yield only a small amount of energy if they are observed. This yields the conclusion that, these states do not contribute much to the input-output behavior of the system, and hence can be neglected.

In general, there are two steps to obtain a reduced model via balanced truncation. The first step is to transform the system to be reduced into a balanced realization, where each state is equally controllable and observable, i.e., the controllable and observable Gramians become equal and diagonal.

Then, the next step is performed by *truncating* the state variables that correspond to smallest singular values $\sigma_j << \sigma_i, i = 1, ..., r, j = r+1, ..., n$. This is achieved numerically by, e.g., the square root balancing method [130] which is summarized in Algorithm 1.

Algorithm 1 Generalized Square Root (GSR) Method

Input: A realization $\mathbf{\Sigma} = \{\mathbf{E}, \mathbf{A}, \mathbf{B}, \mathbf{C}\}$

Output: $\{\mathbf{E}_r\, \mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r\}$

1: Compute the *cholesky factors* $\mathbf{R}_p$ and $\mathbf{L}_p$ of the Gramians $\mathcal{P}$ and $\mathcal{Q}$ that satisfy $\mathcal{P} = \mathbf{R}_p\mathbf{R}_p^T$ and $\mathcal{Q} = \mathbf{L}_p\mathbf{L}_p^T$.

2: Obtain the SVD of $\mathbf{L}_p^T\mathbf{E}\mathbf{R}_p = \mathbf{U}\mathbf{S}\mathbf{V}^T$ and partition:

$$\mathbf{L}_p^T\mathbf{E}\mathbf{R}_p = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{S}_1 & \\ & \mathbf{S}_2\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^T \\ \mathbf{V}_2^T\end{bmatrix},$$

where $\mathbf{U}_1 \in \mathbb{R}^{n\times r}, \mathbf{V}_1^T \in \mathbb{R}^{n\times r}$ are orthogonal, and $\mathbf{S}_1 = \mathrm{diag}(\sigma_1, ..., \sigma_r) \in \mathbb{R}^{r\times r}$.

3: Select $\mathcal{V} = \mathbf{R}_p\mathbf{V}_1\mathbf{S}_1^{-1/2} \in \mathbb{R}^{n\times r}$ and $\mathcal{W}^T = \mathbf{S}_1^{-1/2}\mathbf{U}_1^T\mathbf{L}_p^T \in \mathbb{R}^{r\times n}$ and projection matrices $\mathbf{T} = \mathcal{W}^T$ and $\mathbf{T}^{-1} = \mathcal{V}$.

4: Compute the reduced-order system:

$$\{\mathbf{E}_r, \mathbf{A}_r, \mathbf{B}_r, \mathbf{C}_r\} = \{\mathbf{T}\mathbf{E}\mathbf{T}^{-1}, \mathbf{T}\mathbf{A}\mathbf{T}^{-1}, \mathbf{T}\mathbf{B}, \mathbf{C}\mathbf{T}^{-1}\}.$$

In order to reduce the order of descriptor system (2.22), we have to compute the *Cholesky* factors of the reachability and observability Gramians that satisfy the Lyapunov equation (2.40). These factors can be determined by the generalized *Schur-Hammarling method* [281, 282] without having to compute the solution of Lyapunov equations explicitly. Lyanupov balancing can be applied efficiently for small and moderate size systems, however, in large-scale settings, *exact balancing* becomes highly expensive as it requires dense matrix factorizations resulting in a computational complexity of $\mathcal{O}(n^3)$ and storage of $\mathcal{O}(n^2)$. Towards this direction, approximate balanced reduction methods were proposed, see, e.g., Penzel [230], Antoulas and Sorensen [277], Gugercin et al. [144].

Besides the Lyapunov balancing, there exist other types of balancing methods such as stochastic balancing [92], positive real balancing, bounded real balancing [221], frequency weighted balancing [102], and LQG balancing methods [161]. Furthermore, balanced truncation methods have also been extended to differential-algebraic equations [283, 202, 56], time-varying systems [271, 258, 181], nonlinear systems [267, 50], second-order systems [206, 76, 247], and parametric systems [274].

One of the main advantage of balanced truncation methods is that it generally preserves the asymptotic stability in the reduced model with a rigorous a-priori error bound [130] given as:

$$\left\|\mathbf{G}-\mathbf{G}_r\right\|_{\mathcal{H}_\infty} \leq 2\sum_{1=r+1}^{n}\sigma_i, \tag{2.44}$$

where $\mathbf{G}_r(s) = \mathbf{C}_r(s\mathbf{E}_r - \mathbf{A}_r)^{-1}\mathbf{B}_r$ is the reduced transfer function model. However, the global error bound (2.44) holds only when the Lyapunov equations are solved in exact arithmetic. When using an approximate solution of the Lyapunov equation, the bound is expected to hold only approximately [38, 143]. Despite these issues, balanced truncation still remains the popular MOR choice for LTI systems.

B. Methods based on Krylov subspaces

Another class of model reduction for LTI systems is based on Krylov subspaces. These are also known in MOR literature as *moment-matching* methods [14, 19, 16, 18]. These methods produce a reduced-order model that interpolates the full-order system's response at some predefined interpolation points. The main reason these methods are so successful is their numerical stability and that these methods involve less dense matrix transformations. Some excellent reviews can be found in Refs.[40, 31, 116, 109, 120, 117].

For LTI systems considered here, the transfer function $\mathbf{G}(s)$ can also be written as:

$$\mathbf{G}(s) = -\mathbf{C}(\mathbf{I} - s\mathbf{A}^{-1}\mathbf{E})^{-1}\mathbf{A}^{-1}\mathbf{B}. \tag{2.45}$$

Using the *Neumann* expansion, we can rewrite the above expression as:

$$(\mathbf{I} - s\mathbf{A}^{-1}\mathbf{E})^{-1} = \sum_{i=0}^{\infty}(\mathbf{A}^{-1}\mathbf{E}s)^i, \tag{2.46}$$

which can be further expanded via Taylor series as

$$\begin{aligned}\mathbf{G}(s) = &-\mathbf{C}\mathbf{A}^{-1}\mathbf{B} - \mathbf{C}\mathbf{A}^{-1}\mathbf{E}\mathbf{A}^{-1}\mathbf{B}s - ... \\ &- \mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^i\mathbf{A}^{-1}\mathbf{B}s^i - ...\end{aligned} \tag{2.47}$$

Definition 2.10 (Moments of an LTI system) [14] The moments of the LTI system (2.22) about the expansion point $s = 0$ are the negative coefficients of the Taylor series expression (2.47) expanded about $s = 0$, and are given as

$$\boldsymbol{\eta}_i = \mathbf{C}(\mathbf{A}^{-1}\mathbf{E})^i\mathbf{A}^{-1}\mathbf{B} \qquad i = 0, 1, ... \tag{2.48}$$

The moments of system (2.22) at the point $s = 0$, are also the successive derivatives of the transfer function $\mathbf{G}(s)$ i.e.,

$$\boldsymbol{\eta}_i^0 = -\frac{1}{i}\frac{d^i\mathbf{G}(s)}{ds^i}\bigg|_{s=0} \tag{2.49}$$

If the expansion is performed at some other point $s = \sigma$, then the moments of (2.22) are computed by replacing the matrix $\mathbf{A}$ with $(\mathbf{A} - \sigma\mathbf{E})$ in (2.48) i.e.,

$$\boldsymbol{\eta}_i^\sigma = \mathbf{C}((\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{E})^i(\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{B}, \tag{2.50}$$

for $i = 0, 1, ...$ and where it is assumed that the matrix pencil $(\mathbf{A} - \sigma\mathbf{E})$ is nonsingular. ▲

Definition 2.11 (Markov parameters) [14] The coefficients of the Taylor series of transfer function $\mathbf{G}(s)$ about $s = \sigma$ for $\sigma \to \infty$, are defined as the Markov parameters of the LTI system (2.22) and are defined as:

$$\mathbf{M}_i = \mathbf{C}(\mathbf{E}^{-1}\mathbf{A})^i\mathbf{E}^{-1}\mathbf{B} \qquad i = 0, 1, ... \tag{2.51}$$

Furthermore, the i-th Markov parameter is equal to i-th derivative of the impulse response of system (2.22) at $t = 0$ i.e,

$$\mathbf{M}_i = \frac{d^i\hbar(t)}{dt^i}\bigg|_{t=0} \tag{2.52}$$

This implies that the first Markov parameter $\mathbf{M}_0$ is the system's impulse response at $t = 0$. ▲

2.2.6 Moment-matching in frequency domain

Model reduction via moment matching implies that a reduced-order model is to be obtained whose moments match the the full-order model moments at certain frequencies of interest. If the moments are matched about $s = 0$, the reduced-order model is called the *Padé approximant*, and the problem is known as *Padé's approximation*. If the moments are matched at some other point $s = \sigma$, the problem is known as *shifted Padé's approximation*, and when the Markov parameters are matched, the problem is termed as *partial realization*. It is often desired to match moments at more than one expansion point i.e., at specific frequency intervals, then we talk about the *multipoint Padé* or the *rational interpolation* problem. Thus, model reduction via moment matching is to construct a reduced model (2.38) such that the moments of $\mathbf{G}(s)$ match the reduced system's moments having transfer function $\mathbf{G}_\mathbf{r}(s)$. Now, this begs the questions on how to select the expansion points and, as such, obtain the projection matrices $\mathbf{V}$ and $\mathbf{W}$. This is explained next for both SISO and MIMO case.

SISO case

For SISO systems, (2.22) simplifies to

$$\Sigma_{\text{SISO}} : \begin{cases} \mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{b}\mathbf{u}(t), \\ \mathbf{y}(t) = \mathbf{c}^T\mathbf{x}(t), \end{cases} \tag{2.53}$$

and its ROM via projection is given as:

$$\begin{aligned} \mathbf{E}_r\dot{\mathbf{x}}_r(t) &= \mathbf{A}_r\mathbf{x}_r(t) + \mathbf{b}_r\mathbf{u}(t), \\ \mathbf{y}_r(t) &= \mathbf{c}_r^T\mathbf{x}_\mathbf{r}(t), \end{aligned} \tag{2.54}$$

where

$$\mathbf{E}_r = \mathbf{W}^T\mathbf{E}\mathbf{V}, \mathbf{A}_r = \mathbf{W}^T\mathbf{A}\mathbf{V}, \mathbf{b}_r = \mathbf{W}^T\mathbf{b}, \mathbf{c}_r = \mathbf{c}^T\mathbf{V}, \tag{2.55}$$

and $\mathbf{W}^T\mathbf{b}, \mathbf{c}^T\mathbf{V} \in \mathbb{R}^r$.

Theorem 2.2 (One-sided Padé approximation) *[14, 119, 137] In order to match r moments between the full-order system (2.53) and reduced model (2.54) at the expansion point $s = 0$, it is required that the columns of projection matrix $\mathbf{V}$ used in (2.55) form a basis for the Krylov subspace $\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{b})$. Furthermore the matrix $\mathbf{W}$ is selected such that the matrix $\mathbf{A_r}$ is nonsingular.* ▲

The subspace $\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{b})$ is known as *input Krylov subspace* and the reduced scheme is known as the *one-sided Krylov subspace method*. A typical choice of one-sided Krylov subspace method is $\mathbf{W} = \mathbf{V}$. This has an advantage of preserving stability and passivity of reduced model for some specific large-scale models.

Theorem 2.3 (Two-sided Padé approximation) *[14, 119, 137] In order to match $2r$ moments between the full-order system (2.53) and reduced model (2.54) at the expansion point $s = 0$, it is required that the columns of projection matrices $\mathbf{V}$and $\mathbf{W}$ used in (2.55) form the basis for the Krylov subspaces given as $\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{b})$ and $\mathcal{K}_r(\mathbf{A}^{-T}\mathbf{E}^T, \mathbf{A}^{-T}\mathbf{c})$ respectively, where $\mathbf{A}$ and $\mathbf{A_r}$ are assumed to be invertible.* ▲

This reduction scheme is known as *two-sided Krylov subspace method*, and this corresponds to matching double the number of moments than the one-sided method.

Theorem 2.4 (Shifted Padé approximation) *[14, 119, 137] In order to match $2r$ moments between the full-order system (2.53) and reduced model (2.54) at the expansion point $s = \sigma$, it is required that the columns of projection matrices $\mathbf{V}$and $\mathbf{W}$ used in (2.55) form the basis for the Krylov subspaces given as $\mathcal{K}_r((\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{E}, (A - \sigma\mathbf{E})^{-1}\mathbf{b})$ and $\mathcal{K}_r((\mathbf{A} - \sigma\mathbf{E})^{-T}\mathbf{E}^T, (\mathbf{A} - \sigma\mathbf{E})^{-T}\mathbf{c})$ respectively, where $\mathbf{A}$ and $\mathbf{A_r}$ are assumed to be invertible.* ▲

Remark 2.4 (Shifted approximation) This is achieved by substituting the matrix $\mathbf{A}$ by $(\mathbf{A} - \sigma\mathbf{E})$ in the respective Krylov subspaces as explained earlier.

Sometimes, one is interested in capturing the high-speed dynamics of the system at hand, which is achieved by matching moments at higher frequencies ($s \to \infty$) i.e, matching some of the Markov parameters.

Theorem 2.5 (One-sided Padé and Markov approximation) *Let $m_1 \in \mathbb{Z}$ and $0 \leq m_1 \leq r$, then by selecting the matrix $\mathbf{V}$ used in (2.55) as a basis for Krylov subspace given as $\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, (\mathbf{E}^{-1}\mathbf{A})^{m_1}\mathbf{A}^{-1}\mathbf{b})$ matches the first m_1 Markov parameters and the first $r - m_1$ moments of systems (2.53) and (2.54) resp.* ▲

Similarly, the matrix $\mathbf{W}$ is chosen such that $\mathbf{A}_r$ and $\mathbf{E}_r$ are non-singular. It immediately follows that the number of moments, in this case, can be doubled by selecting the suitable input and output Krylov subspaces.

Theorem 2.6 (Two-sided Padé and Markov approximation) *Let $m_1, m_2 \in \mathbb{Z}$ where $0 \leq m_1, m_2 \leq r$, then by selecting the matrices $\mathbf{V}$ and $\mathbf{W}$, used in (2.55), as the basis for Krylov subspaces given as $\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, (\mathbf{E}^{-1}\mathbf{A})^{m_1}\mathbf{A}^{-1}\mathbf{b})$ and $\mathcal{K}_r(\mathbf{A}^{-T}\mathbf{E}^T, (\mathbf{E}^{-T}\mathbf{A})^{m_2}\mathbf{A}^{-T}\mathbf{c})$ respectively, matches the first $m_1 + m_2$ Markov parameters and the first $2r - m_1 - m_2$ moments of system (2.53) and (2.54) respectively.* ▲

Now, if the aim is to match moments at multiple expansion points: $\sigma_1, \sigma_2, ..., \sigma_k$, then k different Krylov subspaces are to be constructed. In this case, the projection matrix is obtained by the union of all the respective Krylov subspaces such that a universal basis is found.

Theorem 2.7 (Rational Interpolation) *The first r_i moments about σ_i are matched between systems (2.53) and (2.54) respectively, by selecting the matrix $\mathbf{V}$, used in (2.55), as follows:*

$$\bigcup_{i=1}^{k} \mathcal{K}_{r_i}((\mathbf{A} - \sigma_i\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - \sigma_i\mathbf{E})^{-1}\mathbf{b}) \subseteq colspan(\mathbf{V}), \tag{2.56}$$

where its assumed that $(\mathbf{A} - \sigma_i\mathbf{E})$ and $(\mathbf{A_r} - \sigma_i\mathbf{E_r})$ are both nonsingular and $\mathbf{W}$ is an arbitrary full rank matrix. ▲

Similar to previous cases, a two-sided method can provide double the number of moments about each point σ_i. All the theorems mentioned above are summed up in Table 2.1.

Table 2.1: Various examples of MOR by moment matching in SISO systems

Name of reduced order system	Expansion point used	Type of projection	Choice of *Krylov* subspace	Moments matched
Padé[14, 119, 137]	$s = 0$	one-sided	$\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{b})$	r
		two-sided	$\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{b})$, $\mathcal{K}_r(\mathbf{A}^{-T}\mathbf{E}^T, \mathbf{A}^{-T}\mathbf{c})$	$2r$
Partial realization/ *Padé* at ∞/Markov	$s = \infty$	one-sided	$\mathcal{K}_r(\mathbf{E}^{-1}\mathbf{A}, \mathbf{E}^{-1}\mathbf{b})$	r
		two-sided	$\mathcal{K}_r(\mathbf{E}^{-1}\mathbf{A}, \mathbf{E}^{-1}\mathbf{b})$, $\mathcal{K}_r(\mathbf{A}^T\mathbf{E}^{-T}, \mathbf{c}^T)$	$2r$
Shifted *Padé*[14, 137]	$s = \sigma$	one-sided	$\mathcal{K}_r((\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{b})$	r
		two-sided	$\mathcal{K}_r((\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - \sigma\mathbf{E})^{-1}\mathbf{b})$, $\mathcal{K}_r((\mathbf{A} - \sigma\mathbf{E})^{-T}\mathbf{E}^T, (\mathbf{A} - \sigma\mathbf{E})^{-T}\mathbf{c})$	$2r$
Padé and *Markov*	$s = 0$ and $s = \infty$	one-sided	$\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, (\mathbf{E}^{-1}\mathbf{A})^{m_1}\mathbf{A}^{-1}\mathbf{b})$	m_1 at ∞, $r - m_1$ at 0
		two-sided	$\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, (\mathbf{E}^{-1}\mathbf{A})^{m_1}\mathbf{A}^{-1}\mathbf{b})$, $\mathcal{K}_r(\mathbf{A}^{-T}\mathbf{E}^T, (\mathbf{E}^{-T}\mathbf{A})^{m_2}\mathbf{A}^{-T}\mathbf{c})$	$m_1 + m_2$ at ∞, $2r - m_1 - m_2$ at 0
Rational/ Multipoint *Padé*	$s_i = \sigma_1, \sigma_2, ..., \sigma_k$	one-sided	$\bigcup_{i=1}^{k} \mathcal{K}_{r_i}((\mathbf{A} - \sigma_i\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - \sigma_i\mathbf{E})^{-1}\mathbf{b})$	r_i about σ_i
		two-sided	$\bigcup_{i=1}^{k} \mathcal{K}_{r_i}((\mathbf{A} - \sigma_i\mathbf{E})^{-1}\mathbf{E}, (\mathbf{A} - \sigma_i\mathbf{E})^{-1}\mathbf{b})$, $\bigcup_{i=1}^{k} \mathcal{K}_{r_i}((\mathbf{A} - \sigma_i\mathbf{E})^{-T}\mathbf{E}^T, (\mathbf{A} - \sigma_i\mathbf{E})^{-T}\mathbf{c})$	$2r_i$ about σ_i

So far, it is not clear how to numerically obtain the projection matrices $\mathbf{V}$ and $\mathbf{W}$. An early attempt at this was the Asymptotic Waveform Evaluation (AWE) method [233] in which the moments were explicitly calculated rather than computing the matrices $\mathbf{V}$ and $\mathbf{W}$. The method became prominent due to its capability to reduce RC interconnect models containing thousands of variables. Later on, a multi-point version of the method was proposed by Chiprout and Nakhla [84]. AWE-based methods, however, had a numerical instability that the vectors become linearly dependent and converge to an eigenvector of $\mathbf{A}$. This shortcoming was first pointed out by Gallivan et al. [124] and later by Feldman and Freund [107]. Consequently, this led to the development of implicit-based moment-matching methods, and the first significant contribution came via the *Arnoldi* method [22, 117]. This method employs a one-sided projection to iteratively construct a set of normalized vectors satisfying $\mathbf{V}^T\mathbf{V} = \mathbf{I}$. Within every iteration, a new vector is generated, which is orthogonal to all the previous ones. This results in an upper Hessenberg structure of the matrix $\mathbf{A_r}$ and the vector $\mathbf{b_r}$ becomes a multiple of the first unit vector. However, this algorithm also generates a linearly dependent set of basis vectors for a relatively large value of r. This is normally avoided by deflating the redundant columns of $\mathbf{V}$ to retain reduced models with a certain degree of accuracy [89]. This has been demonstrated by B. Salimbahrami [256] where a modified Gram-Schmidt orthogonalization scheme is employed.

Another key contribution towards implicit moment matching was the two-sided *Lanczos* method [107, 180], also known as the *Padé via Lanczos (PVL)* method. This method constructs two sequences of basis vectors which span the respective input and output Krylov subspaces satisfying $\mathbf{W}^T\mathbf{V} = \mathbf{I}$, resulting in an upper triangular structure of the matrix $\mathbf{A_r}$. Apart from matching moments, this method

was initially proposed for obtaining reduced models based on the computation of eigenvalues [219]. Further work in this direction led to the development of partial realization via Lanczos method by Gragg and Lindquist [132]. The method was also extended to MIMO systems by Aliaga et al. [4] (also see Refs. [167, 168]). Boley [59] addressed the issue regarding the loss of biorthogonalization in the classical Lanczos method. Later-on, new results also appeared in the areas of stability retention [138], error analysis [159] and in control literature [59, 292]. However, these studies didn't present any new structure in projection technique for rational interpolation.

To retain passivity among the reduced models, the passive reduced-order interconnect macromodelling algorithm (PRIMA) was proposed by Odabasioglu et al. [220]. The idea was demonstrated in linear RLC systems. In order to match moments at multiple expansion points, the rational Lanczos method and the dual Arnoldi method were proposed by Grimme et al. [137]. Furthermore, the issue of unstable partial realizations in classical Krylov methods was addressed by the restarting techniques proposed by Grimme et al.[138], and the implicitly restarted dual Arnoldi method by Jaimoukha and Kasemally [160].

MIMO case

For the case of MIMO systems, (given in (2.22)), the *block Krylov subspaces* are defined as:

$$\mathcal{K}_r(\mathbf{A}, \mathbf{B}) = span(\mathbf{B}, \mathbf{AB}, ..., \mathbf{A}^{r-1}\mathbf{B}). \tag{2.57}$$

The block subspace for m starting vectors/columns of $\mathbf{B}$ can be considered as a union of m Krylov subspaces for each starting vector [256]. Thus, by using the block Krylov subspaces, all the above mentioned theorems for matching the moments/Markov parameters can be generalized for MIMO systems. For instance, one has to use the block versions, i.e, $\mathcal{K}_r(\mathbf{A}^{-1}\mathbf{E}, \mathbf{A}^{-1}\mathbf{B})$ and $\mathcal{K}_r(\mathbf{A}^{-T}\mathbf{E}^T, \mathbf{A}^{-T}\mathbf{C}^T)$ that generalizes Theorem 2.3. Consequently, $\frac{r}{m}$ moments are matched in one-sided method and $\frac{r}{m} + \frac{r}{p}$ in the two-sided method.

Tangential Interpolation problem

The notion of interpolation for MIMO systems implies that the interpolating matrix-valued rational function $\mathbf{G}_r(s)$ matches the origin function $\mathbf{G}(s)$ with respect to some predefined error. This would, in effect, require $p \times m$ interpolation conditions for each interpolation point. As such, this will result in large size of reduced-order model r for even a moderate size of input and output dimensions m and p. Thus, for MIMO systems, it's desired that the interpolating function matches the original function along certain specific directions or tangent directions. Consequently, this involves selecting interpolation points as well as the interpolation directions.

Tangential interpolation based moment matching for MIMO systems has been presented in many studies such as in Ref's. [14, 17, 31, 223, 107, 119, 124]. It is the extension of rational Krylov methods to MIMO systems when all tangent directions are the same. The tangential interpolation problem for descriptor systems was also proposed by Gugercin et al. [145] and for index one descriptor system by Antoulas et al. [17]. In the following, we brief the tangential interpolation problem.

Definition 2.12 (Right-tangent interpolant) Let $\mathbf{w}_i \in \mathbb{C}^m$ be the (non-trivial) right tangent direction, we define $\mathbf{G}_r(s)$ to be the right tangent interpolant of $\mathbf{G}(s)$ at $s = \sigma_i$ along $\mathbf{w}_i$ if

$$\mathbf{G}(\sigma_i)\mathbf{w}_i = \mathbf{G}_r(\sigma_i)\mathbf{w}_i. \tag{2.58}$$

▲

Similarly, the left tangential interpolant is defined as:

Definition 2.13 (Left-tangent interpolant) Let $\mathbf{v}_i \in \mathbb{C}^p$ be the (non-trivial) left tangent direction, then we define $\mathbf{G}_r(s)$ to be the left tangent interpolant of $\mathbf{G}(s)$ at $s = \mu_i$ along $\mathbf{v}_i$ if

$$\mathbf{v}_i^T\mathbf{G}(\mu_i) = \mathbf{v}_i^T\mathbf{G}_r(\mu_i). \tag{2.59}$$

▲

Thus, given a set of r left interpolation points $\{\mu_i\}_{i=1}^r$, r left tangential directions $\{\mathbf{v}_i\}_{i=1}^r$, r right interpolation points $\{\sigma_i\}_{i=1}^r$, and r right tangential directions $\{\mathbf{w}_i\}_{i=1}^r$, we can formulate the model reduction problem via tangential interpolation as finding a degree-r reduced transfer function $\mathbf{G}_r(s)$ such that (2.58) and (2.59) hold for $i = 1, 2, .., r$.

Definition 2.14 (Bitangential Hermite interpolant) Given a set of r left and r right tangential directions as $\mathbf{v}_i$ and $\mathbf{w}_i$ respectively, we define $\mathbf{G}_r(s)$ to be a bitangential Hermite interpolant of $\mathbf{G}(s)$, if $\mathbf{G}_r(s)$ satisfies both (2.58) and (2.59). In addition to this, it is required that

$$\mathbf{v}_i^T \frac{d\mathbf{G}(s)}{ds}\bigg|_{s=\sigma_i} \mathbf{w}_i = \mathbf{v}_i^T \frac{d\mathbf{G}_r(s)}{ds}\bigg|_{s=\sigma_i} \mathbf{w}_i, \tag{2.60}$$

holds for $i = 1, 2, ..., r$. ▲

Theorem 2.8 (Tangential interpolation) *[125] If the columns of matrices* $\mathbf{V}$ *and* $\mathbf{W}$ *used in (2.55) are selected as:*

$$(\sigma\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{w} \in Ran(\mathbf{V}), \tag{2.61a}$$

$$(\mathbf{v}^T\mathbf{C}(\mu\mathbf{E} - \mathbf{A})^{-1})^T \in Ran(\mathbf{W}), \tag{2.61b}$$

then the following tangential interpolation conditions are satisfied

$$\mathbf{G}(\sigma)\mathbf{w} = \mathbf{G}_r(\sigma)\mathbf{w}, \tag{2.62a}$$
$$\mathbf{v}^T\mathbf{G}(\mu) = \mathbf{v}^T\mathbf{G}_r(\mu), \tag{2.62b}$$

with the assumption that $(\sigma\mathbf{E} - \mathbf{A})$ *and* $(\mu\mathbf{E} - \mathbf{A})$ *are invertible. Furthermore, if (2.62) hold with* $\sigma = \mu$ *then the bitangential Hermite condition, i.e,*

$$\mathbf{v}^T \frac{d\mathbf{G}(s)}{ds}\bigg|_{s=\sigma} \mathbf{w} = \mathbf{v}^T \frac{d\mathbf{G}_r(s)}{ds}\bigg|_{s=\sigma} \mathbf{w}, \tag{2.63}$$

is satisfied as well. ▲

Theorem 2.8 demonstrates a left or right interpolation condition without the need to explicitly calculate the values that are interpolated. The idea can be easily extended to the case of r interpolation points i.e., given a set of r left interpolation points $\{\mu_i\}_{i=1}^r$, r right interpolation points $\{\sigma_i\}_{i=1}^r$, r left tangent directions $\{\mathbf{v}_i\}_{i=1}^r$ and r right tangent directions $\{\mathbf{v}_i\}_{i=1}^r$ respectively, we can construct the matrices $\mathbf{V}$ and $\mathbf{W}$, that satisfy the Lagrange tangential interpolation conditions in (2.58,2.59) and also the bitangential Hermite interpolation condition in (2.60) for $\sigma_i = \mu_i$, given as follows:

$$[(\sigma_1\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{w}_1, ..., (\sigma_r\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{w}_r] = \mathbf{V}, \tag{2.64}$$

and

$$[(\mu_1\mathbf{E} - \mathbf{A})^{-T}\mathbf{C}^T\mathbf{v}_1, ..., (\mu_r\mathbf{E} - \mathbf{A})^{-T}\mathbf{C}^T\mathbf{v}_r] = \mathbf{W}. \tag{2.65}$$

The scheme can also be used for higher-order Hermite interpolation given as follows:

Theorem 2.9 (Higher-order Hermite interpolation) *[125] Given the interpolation points* $\sigma, \mu \in \mathbb{C}$ *and the (nontrivial) tangent directions* $\mathbf{v} \in \mathbb{C}^p$ *and* $\mathbf{w} \in \mathbb{C}^m$*, let the matrix pencils* $(\sigma\mathbf{E} - \mathbf{A})$ *and* $(\mu\mathbf{E} - \mathbf{A})$ *be invertible, then if the columns of matrix* $\mathbf{V}$ *used in (2.55) are constructed as*

$$((\sigma\mathbf{E} - \mathbf{A})^{-1}\mathbf{E})^{k-1}(\sigma\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}\mathbf{w} \in Ran(\mathbf{V}), \tag{2.66}$$

for $k = 1, .., N$*, then*

$$\frac{d^j\mathbf{G}(s)}{ds^j}\bigg|_{s=\sigma} \mathbf{w} = \frac{d^j\mathbf{G}_r(s)}{ds^j}\bigg|_{s=\sigma} \mathbf{w} \quad \textit{for } j = 0, .., N-1, \tag{2.67}$$

and if the columns of matrix $\mathbf{W}$ *used in (2.55) are constructed as*

$$((\mu\mathbf{E} - \mathbf{A})^{-T}\mathbf{E}^T)^{k-1}(\mu\mathbf{E} - A)^{-T}\mathbf{C}^T\mathbf{v} \in Ran(\mathbf{W}), \tag{2.68}$$

for $k = 1, .., M$, *then*

$$\mathbf{v}^T \frac{d^j \mathbf{G}(s)}{ds^j}\bigg|_{s=\mu} = \mathbf{v}^T \frac{d^j \mathbf{G}_r(s)}{ds^j}\bigg|_{s=\mu}; j = 0, .., M-1, \tag{2.69}$$

and if both (2.67) and (2.69) hold with $\sigma = \mu$ *then*

$$\mathbf{v}^T \frac{d^j \mathbf{G}(s)}{ds^j}\bigg|_{s=\sigma} \mathbf{w} = \mathbf{v}^T \frac{d^j \mathbf{G}_r(s)}{ds^j}\bigg|_{s=\sigma} \mathbf{w}, \tag{2.70}$$

for $j = 0, ..., N + M - 1$. ▲

The main benefit of interpolatory model reduction methods is that it avoids solving large-scale Lyapunov or Riccati equations, making it a convenient reduction platform for large-scale systems. However, the main cost among these methods is solving the shifted linear systems that are sparse in most cases and can be solved using direct methods (e.g., Gaussian elimination). Also, one can prefer iterative solution methods for obtaining the reduction basis $\mathbf{V}$ and $\mathbf{W}$ when dealing with systems with millions of degrees of freedom.

2.2.7 Moment-matching in time-domain

So far, we have discussed the frequency domain notion of moment matching for LTI systems for both SISO and MIMO systems. We saw that the reduced model obtained using Krylov subspaces results in a local approximation of frequency response. However, this cannot guarantee a good overall approximation of impulse response. Gunupudi and Nakhla [146], were among the first to present a scheme to match the first derivatives of time-response of full order model with that of the reduced one. Later on, Wang et al. [301] presented a passive model order reduction method based on Chebyshev's expansion of the impulse response to match the transient responses of the full model and the reduced-order model. The Laguerre polynomial expansion-based reduction framework was presented by Chen et al. [83]. This was later extended and further developed by Rudy [100]. However, the major contribution in this direction came from Astolfi [24] who also extended the idea further to nonlinear systems [25, 26, 27]. In the following, we describe the time-domain notion of moment matching based on Refs. [24, 25, 26].

Definition 2.15 (Time-domain moments of LTI systems) The moments $\bar{\boldsymbol{\eta}}_{\boldsymbol{i}}(\sigma)$ of the impulse response $\mathbf{h}(t)$ around a point $s = \sigma$ is given by the weighted integrals over the time function and satisfy

$$\bar{\boldsymbol{\eta}}_{\boldsymbol{i}}(\sigma) = \int_0^\infty \tau^i e^{-\sigma\tau} \mathbf{h}(t) d\tau, \tag{2.71a}$$

$$= \int_0^\infty \tau^i \mathbf{C} e^{(\mathbf{E}^{-1}\mathbf{A} - \sigma\mathbf{I})\tau,} \mathbf{E}^{-1}\mathbf{B} d\tau \tag{2.71b}$$

$$= i!\ \mathbf{C}((\sigma\mathbf{E} - \mathbf{A})^{-1}\mathbf{E})^i (\sigma\mathbf{E} - \mathbf{A})^{-1}\mathbf{B}. \tag{2.71c}$$

▲

Remark 2.5 Thus, the time-domain moments $\bar{\boldsymbol{\eta}}_{\boldsymbol{i}}(\sigma)$ and the frequency-domain moments $\boldsymbol{\eta}_{\boldsymbol{i}}(\sigma)$ only differ by a factor:

$$\boldsymbol{\eta}_{\boldsymbol{i}}(\sigma) = \frac{(-1)^i}{i!}\bar{\boldsymbol{\eta}}_{\boldsymbol{i}}(\sigma). \tag{2.72}$$

This result is used to develop the notion of moments in terms of steady-state response of system (2.22) interconnected with a linear signal generator. ▲

Definition 2.16 (Moment in time-domain) *[23]* The 0-th moment of system (2.22) at $s = \sigma$ can also be defined as

$$\boldsymbol{\eta}_{\mathbf{0}}(\sigma) = \mathbf{CV}, \tag{2.73}$$

where $\mathbf{V} \in \mathbb{R}^{n\times r}$ uniquely solves the linear Sylvester equation given as:

$$\mathbf{AV} + \mathbf{B} = \sigma\mathbf{EV}. \tag{2.74}$$

▲

Thus, the moments of system (2.22), $\boldsymbol{\eta}_0(\sigma), \boldsymbol{\eta}_1(\sigma), ..., \boldsymbol{\eta}_k(\sigma)$ can be uniquely determined by the elements of matrix $\mathbf{CV}$ i.e., there exists a one-to-one relation between the moments of system (2.22) and the elements of matrix $\mathbf{CV}$, where $\mathbf{V}$ being the unique solution of the following Sylvester equation

$$\mathbf{EV\Xi} - \mathbf{AV} = \mathbf{B\Psi}, \tag{2.75}$$

where $\mathbf{\Xi} \in \mathbb{R}^{r\times r}$ is any non-derogatory matrix, and it is assumed that the pair $(\mathbf{\Psi}, \mathbf{\Xi})$ is observable.
Now, consider the following exogenous linear system (also known as signal generator):

$$\dot{\boldsymbol{\zeta}}(t) = \mathbf{\Xi}\boldsymbol{\zeta}(t), \qquad \boldsymbol{\zeta}(0) = \boldsymbol{\zeta}_{\mathbf{0}} \neq 0, \tag{2.76a}$$
$$\mathbf{u}(t) = \mathbf{\Psi}\boldsymbol{\zeta}(t), \tag{2.76b}$$

with $\boldsymbol{\zeta}(t) \in \mathbb{R}^r$, $\lambda(\mathbf{E}^{-1}\mathbf{A}) \cap \lambda(\mathbf{\Xi}) = \emptyset$ and the triple $(\mathbf{\Xi}, \mathbf{\Psi}, \boldsymbol{\zeta}(0))$ to be minimal i.e., the pair $(\mathbf{\Psi}, \mathbf{\Xi})$ is observable and $(\mathbf{\Xi}, \boldsymbol{\zeta}(0))$ is controllable such that the generated input signal $\mathbf{u}(t)$ is persistently exciting [29, 188].

Theorem 2.10 (Steady-state matching via moments) *[25, 26] Consider system (2.22), $\sigma \in \mathbb{C}$. Assume $\lambda(\mathbf{E^{-1}A}) \cap \lambda(\mathbf{\Xi}) = \emptyset$. Let $\mathbf{V}$ satisfies the Sylvester equation (2.75) and $\mathbf{W}$ such that $det(\mathbf{W}^T\mathbf{EV} \neq 0)$. Consider the interconnection of exogenous system (2.76) with system (2.22) as follows:*

$$\mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{Ax}(t) + \mathbf{B\Psi}(e^{\mathbf{\Xi}t}\boldsymbol{\zeta}(\mathbf{0})), \tag{2.77a}$$
$$\mathbf{y}(t) = \mathbf{Cx}(t), \tag{2.77b}$$

where the pair $(\mathbf{\Psi}, \mathbf{\Xi})$ is observable. Then, the (well-defined) steady-state of the output of the said interconnection can be uniquely determined by the respective moments $\boldsymbol{\eta}_0(\sigma), \boldsymbol{\eta}_1(\sigma), ..., \boldsymbol{\eta}_k(\sigma)$. ▲

Thus, connecting a linear signal generator with the full order system (2.22) is similar to providing exponential inputs to the system given as $\mathbf{u}(t) = \boldsymbol{\Psi}(e^{\boldsymbol{\Xi} t}\boldsymbol{\zeta}(\mathbf{0}))$. The (well-defined) steady-state of interconnected system is given as:

$$\mathbf{y}(t) = \underbrace{\mathbf{C}e^{\mathbf{E}^{-1}\mathbf{A}(t)}(\mathbf{x}(0) - \mathbf{V}\boldsymbol{\zeta}(0))}_{\mathbf{y}_p(t)} + \underbrace{\mathbf{C}\mathbf{V}\boldsymbol{\zeta}(t)}_{\mathbf{y}_h(t)} \tag{2.78}$$

where $\mathbf{y}_h(t)$ represents the steady-state response of the system and for an asymptotically stable system, considered here, the transient response, represented by $\mathbf{y}_p(t)$ decays to zero at $t \to \infty$.

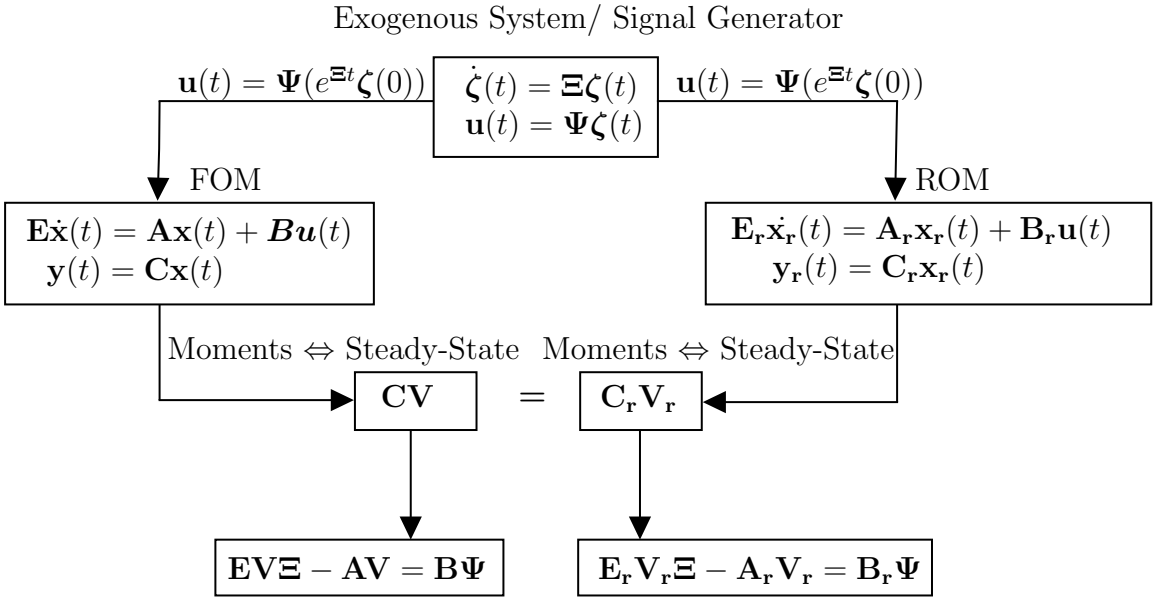

Figure 2.3: Time-domain illustration of linear moment-matching in terms of steady-state response matching

Using Theorem 2.10, the reduced-order model via moment-matching can be obtained as follows:

Theorem 2.11 (Moment-matching in time domain) *[26] Consider the full-order model (2.22) and the system described in (2.38). Fix* $\boldsymbol{\Psi}$ *and* $\boldsymbol{\Xi}$ *such that the pair* $(\boldsymbol{\Psi}, \boldsymbol{\Xi})$ *is observable. Also assume that* $\lambda(\mathbf{E}^{-1}\mathbf{A}) \cap \lambda(\boldsymbol{\Xi}) = \emptyset$ *. Then, the reduced model (2.38) matches moments with system (2.22) at* $(\boldsymbol{\Xi}, \boldsymbol{\Psi}, \boldsymbol{\zeta}(0))$ *if*

$$\mathbf{C}\mathbf{V} = \mathbf{C}_\mathbf{r}\mathbf{V}_\mathbf{r}, \tag{2.79}$$

where $\mathbf{V}_\mathbf{r}$ *is the unique solution of the Sylvester equation given as:*

$$\mathbf{E}_\mathbf{r}\mathbf{V}_\mathbf{r}\boldsymbol{\Xi} - \mathbf{A}_\mathbf{r}\mathbf{V}_\mathbf{r} = \mathbf{B}_\mathbf{r}\boldsymbol{\Psi}. \tag{2.80}$$

▲

Thus, moment-matching in the time-domain implies matching the full order model's steady-state response with the reduced model when both are excited by appropriate inputs from the signal generator. However, the steady-state response is interpolated for inputs other than those generated from the signal generator.

The time-domain illustration of moment-matching in terms of interpolation of steady-state response is depicted in Fig. 2.3. This kind of notion of moment-matching in terms of steady-state response is particularly useful in the sense that as it permits one to define moments even for those systems which don't have a transfer function representation, such as the time-varying systems.

2.2.8 Issues and recent advancements in linear moment-matching

Model order reduction using Krylov subspace methods provides an indispensable platform for reducing large-scale problems; however, there are many issues related to the automatic generation of the reduced model. Although various algorithms have been developed in the past, these apply only to some specific class of problems or under some specific conditions. Next, we highlight some major relevant issues and overview some of the significant achievements in this direction.

A. Choice of expansion point(s)

The choice and number of interpolation points or shifts in Krylov subspace methods is an important factor in dictating the quality of the approximation.To address this issue, a lot of work has been carried out over the past years. In order for the reduced system to minimize the $\mathcal{H}_2$-norm error, various optimality conditions have been formulated either in terms of rational interpolation conditions [42, 43, 41, 66, 140, 141, 172, 203, 293] or in terms of Sylvester and Lyapunov equations [55, 148, 278, 310, 314]. For SISO systems, the interpolation conditions were first proposed by Meier and Luenberger [203]. Gugercin et al. proposed the iterative rational Krylov algorithm (IRKA) [140, 142], which produces reduced models satisfying the first-order necessary conditions for $\mathcal{H}_2$ optimality by selecting the interpolation points as the mirror images of the poles of the reduced system. The idea was later extended to MIMO systems in Refs. [66, 141, 293]. Van Dooren et al. derived the optimality conditions for the case of repeated poles of $\mathbf{G_r}(s)$ [293]. Generally speaking, IRKA has been a significant success in obtaining optimal reduced models, and, as such, finds applications in psychophysiology [165], optimal cooling of steel profiles [141] and many others. Recently, this method was also extended for the reduction of bilinear systems by Benner and Breiten [46]. The convergence of IRKA is guaranteed a priori in some situations [114], albeit fails in some cases [114, 141]. However, the choice of initial starting values remains an open question.

Druskin and Simoncini [96] proposed an adaptive computation of interpolation points for the rational Krylov methods. Though this method is less

accurate than IRKA, it has a low computational cost. This was later followed by the SPARK algorithm by Panzer et al. [225] in which the choice of interpolation point(s), as well as the order of reduced model, is adaptively selected. Similar work was also presented in Ref. [110]. Based on the binary search principle, Bollhöfer and Bodendiek [60] presented adaptive rules for selecting the shifts. They showed a general scheme for determining both the shifts and the moments by combining the adaptive shift selection scheme with the adaptive moment selecting method as presented earlier in Ref. [182]. However, the choice of selecting the reduced-order dimension remains unknown.

Besides the appropriate choice of interpolation point, it is often required in certain applications that the interpolation is carried out in specific frequency regions of interest. This involves weighting certain frequencies more than others. To allow frequency weighting, the weighted $\mathcal{H}_2$ model reduction was first proposed by Halevi [148, 278] as the solution of Riccati and Lyapunov equations. However, a more efficient version of this scheme was introduced in SISO systems by Anić et al. [11] and for MIMO systems by Breiten et al. [63].

All the above methods described demand more or less heuristics, as the user has to manually pick and try several interpolation points and decide what works satisfactorily for their application. This is because of the absence of global error bounds for Krylov subspace methods, which will be discussed next. However, a general scheme that one can follow (as given in Ref. [100]) is that by selecting $s = 0$, the steady-state accuracy is improved as the DC gain of both the full-order and reduced-order model is matched. By selecting $s \to \infty$, the resulting reduced system better approximates the transient response of the full-order system. Also, selecting multiple interpolation points across the frequency spectrum leads to a better approximation on a broader frequency band.

B. Global error indicator

Another major open issue in Krylov-based reduction is the lack of a global error bound between the full-order system and the reduced one. Earlier results in this direction led to the development of local error bounds for the transfer function only for a certain frequency range [34, 137]. Later, some heuristic error indicators were presented in Refs. [44, 137]. It was proposed that the error between the full-order model and the reduced model is approximately equal to one between two successive reduced-order models. However, no proof was provided. A deductible error estimator was presented first by Konkel et al. [169].

A posterior error indicator was presented by Feng, and Benner [108], whereby a greedy algorithm determines the next interpolation point based on the largest error. However, the method is applicable only to some special class

of LTI systems where the matrix $\mathbf{E}$ in (2.22) is symmetric positive definite. A similar requirement is mandated in the method presented by Panzer et al. [226] whereby global $\mathcal{H}_2$ and $\mathcal{H}_\infty$ error bounds are derived. Furthermore, it is also required that $\mathbf{A} + \mathbf{A}^T > \mathbf{0}$. Another study by Wolf et al. [309] proposed a gramian-based output error bound. However, the method is not computationally practical for large-scale systems as it involves the explicit computation of observability gramian.

Thus, we conclude that the calculation of an exact global error bound in Krylov-based reduction methods requires the involvement of the full-order system, and this would become practically challenging for large-scale settings. As such, this remains an open area of research.

C. Preserving Stability and Passivity
One of the most fundamental requirements in any reduction scheme is that the property of stability and passivity remains preserved in the reduced model to make sense of surrogate modeling. It is well-known that Krylov subspace-based reduced-order modeling techniques do not preserve stability or passivity in general. However, there are some appreciable efforts in this direction. Methods proposed in Refs. [118, 166] offer guaranteed stability and passivity of the reduced models if the original system is passive. Another set of methods were presented in Refs.[32, 138, 160]. These methods are based on post-processing schemes, in which the unstable poles of the reduced system are removed using explicitly restarted Lanczos and Arnoldi methods. Inspired by the relation between Löwner and Pick matrices, the interpolation-based passivity preserving methods were proposed in Refs.[15, 276].

Similar to the previous discussion, these methods are confined to a special class of LTI systems and extend only to one-sided Krylov methods. The interpolation-based methods are numerically costly than classical Krylov-based MOR. As far as the restarted algorithms are considered, these apply to SISO systems only. Furthermore, after removing the unstable poles, these methods do not preserve moment matching property and, as such, do not always guarantee to obtain a stable a reduced model with a finite number of restarts, thus making this an open problem to be addressed.

2.3 Chapter summary

In this chapter, we have reviewed some of the mathematical fundamentals involved in MOR. We first discussed some tools from linear algebra that form the basis of this thesis. Then, we formally introduced the MOR in linear systems via projection. We presented an overview of the most popular reduction approaches for LTI systems and pointed out the differences and similarities among various methods. Then, we described the problem of MOR via the moment-matching perspective.

We explained the different Krylov subspaces used to obtain the reduced models for SISO systems. We have also examined the tangential interpolation-based moment-matching methods for MIMO systems. Both frequency and time-domain notions of the moment-matching technique were discussed. Several open problems were also highlighted. In the next chapter, we will extend these system-theoretic concepts to nonlinear systems.

Part II

Nonlinear State-Space Systems

Chapter 3

Model Order Reduction in Nonlinear Systems

In this part of the thesis, we focus on MOR for a general class of nonlinear state-space systems. First, we will discuss the two classes of projection techniques involved in the reduction of nonlinear systems. Afterward, we will present an overview of some well-established nonlinear MOR routines and highlight the similarities and differences among them. Then, we will discuss the nonlinear enhancement of the moment-matching technique in terms of steady-state response matching. At the end of this chapter, we will also discuss the hyper-reduction method for nonlinear systems.

3.1 Nonlinear time-invariant systems

Consider a large-scale, exponentially stable, time-invariant, MIMO, nonlinear system in state-space form as follows:

$$\boldsymbol{\Sigma}_{\mathrm{NL}} \begin{cases} \mathbf{E}\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), & \mathbf{x}(0) = \mathbf{x_0} \\ \mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t)), \end{cases} \tag{3.1}$$

with non-singular descriptor matrix $\mathbf{E} \in \mathbb{R}^{n \times n}$, the vectors $\mathbf{x}(t) \in \mathbb{R}^n, \mathbf{u} \in \mathbb{R}^m, \mathbf{y}(t) \in \mathbb{R}^p$, and two nonlinear, vector valued functions $\mathbf{f}(\mathbf{x}, \mathbf{u}) : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n, \mathbf{g}(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^p$ such that $\mathbf{f}(\mathbf{0}, \mathbf{0}) = \mathbf{0}$ and $\mathbf{g}(\mathbf{0}) = \mathbf{0}$. We assume that the zero equilibrium, (calculated from $\mathbf{0} = \mathbf{f}(\mathbf{x}, \mathbf{0})$), is locally exponentially stable.

Unlike in linear systems, the input-output relationship in a nonlinear system cannot be analytically expressed in terms of transfer functions, convolutional integrals, or state-transition matrix. This is possible only for some special polynomial nonlinear systems such as bilinear or quadratic-bilinear systems via the *Volterra* series representations [252, 62].

3.2 Reduction using projection

Similar to LTI systems, the reduction of the nonlinear state-space model (3.1) is also carried using projection technique. However, for nonlinear systems, we distinguish between the linear and nonlinear Petrov-Galerkin projection.

3.2.1 Nonlinear Petrov-Galerkin projection

One promising way to obtain a reduced-order model of a nonlinear system is by applying a *nonlinear* Petrov-Galerkin projection, whereby the approximation used is as follows:

$$\mathbf{x}(t) = \wp(\mathbf{x}_r(t)) + \mathbf{e}(t), \tag{3.2}$$

where $\wp(\mathbf{x}_r) : \mathbb{R}^r \to \mathbb{R}^n$ is a nonlinear smooth mapping, and $\mathbf{e}(t)$ is the approximation error. The derivative of (3.2) yields

$$\dot{\mathbf{x}}(t) = \tilde{\mathbf{V}}\dot{\mathbf{x}}_r(t) + \dot{\mathbf{e}}(t), \tag{3.3}$$

where

$$\tilde{\mathbf{V}} = \frac{\partial \wp(\mathbf{x}_r(t))}{\partial \mathbf{x}_r(t)} \in \mathbb{R}^{n\times r}, \tag{3.4}$$

is the Jacobian matrix. Substituting the approximation ansatz (3.2), and its derivative (3.3) in (3.1) results in an over-determined system of equations given as:

$$\mathbf{E}\tilde{\mathbf{V}}_{\mathbf{x}_r}\dot{\mathbf{x}}_r(t) - \mathbf{f}(\wp(\mathbf{x}_r(t), \mathbf{u}(t))) - \mathbf{r}(t) = \mathbf{0}. \tag{3.5}$$

with $\mathbf{r}(t) \in \mathbb{R}^n$ as the residual term. The resulting system is then projected onto the subspace $\tilde{\mathcal{U}} = \text{ran}(\mathbf{E}\tilde{\mathbf{V}}_{\mathbf{x}_r})$ to obtain a square ROM. The projection is performed along the orthogonal complement $\tilde{\mathcal{W}}^{\perp} = \text{ran}(\tilde{\mathbf{W}}_{\mathbf{x}_r})^{\perp}$ where

$$\tilde{\mathbf{W}}^T_{\mathbf{x}_r} = \left.\frac{\partial \boldsymbol{\varphi}(\boldsymbol{\omega}(\mathbf{x}, \mathbf{u}))}{\partial \mathbf{x}}\right|_{\mathbf{x}=\wp(\mathbf{x}_r)} \in \mathbb{R}^{r\times n}, \tag{3.6}$$

with the mapping $\boldsymbol{\varphi}(\boldsymbol{\omega}(\mathbf{x}, \mathbf{u})) : \mathbb{R}^n \to \mathbb{R}^r$. Multiplying the overdetermined system (3.5) from the left with the projector $\mathbf{P} = \mathbf{E}\tilde{\mathbf{V}}_{\mathbf{x}_r}(\tilde{\mathbf{W}}^T_{\mathbf{x}_r}\mathbf{E}\tilde{\mathbf{V}}_{\mathbf{x}_r})^{-1}\tilde{\mathbf{W}}^T_{\mathbf{x}_r}$ gives:

$$\mathbf{P}\left(\underbrace{\mathbf{E}\tilde{\mathbf{V}}_{\mathbf{x}_r}\dot{\mathbf{x}}_r(t) - \mathbf{f}(\wp(\mathbf{x}_r(t), \mathbf{u}(t)))}_{\boldsymbol{\omega}\left(\wp(\mathbf{x}_r(t), \mathbf{u}(t))\right)} - \mathbf{r}(t)\right) = \mathbf{0}. \tag{3.7}$$

Finally, enforcing a Petrov-Galerkin condition $\tilde{\mathbf{W}}^T_{\mathbf{x}_r}\mathbf{r}(t) = \mathbf{0}$, vanishes the residual leading to the final ROM of dimension r given as:

$$\begin{aligned}\mathbf{E}_r\dot{\mathbf{x}}_r(t) &= \left.\frac{\partial\boldsymbol{\varphi}(\boldsymbol{\omega}(\mathbf{x}(t),\mathbf{u}(t)))}{\partial\mathbf{x}(t)}\right|_{\mathbf{x}(t)=\wp(\mathbf{x}_r(t))} \mathbf{f}(\wp(\mathbf{x}_r(t),\mathbf{u}(t))), \quad \mathbf{x}_r(0)=\mathbf{x}_{r,0},\\ \mathbf{y}_r(t) &= \mathbf{g}(\wp(\mathbf{x}_r(t))),\end{aligned} \tag{3.8}$$

where $\mathbf{E}_r = \tilde{\mathbf{W}}_{\mathbf{x}_r}^T\mathbf{E}\tilde{\mathbf{V}}_{\mathbf{x}_r}$ and the reduced nonlinear function $\mathbf{f}_r(\mathbf{x}_r(t),\mathbf{u}(\mathbf{t})) = \tilde{\mathbf{W}}_{\mathbf{x}_r}^T\mathbf{f}(\wp(\mathbf{x}_r(t)),\mathbf{u}(t))$.

3.2.2 Linear Petrov-Galerkin projection

Another successful and established approach to reduced nonlinear systems is to apply the classical Petrov-Galerkin projection as discussed for linear systems in Section 2.2.4. Substituting the linear approximation ansatz (2.34) in the FOM (3.1) and premultiplying the overdetermined system with the projector $\mathbf{P}$ given in (2.36) yields:

$$\mathbf{P}(\underbrace{\mathbf{E}\mathbf{V}\dot{\mathbf{x}}_r(t) - \mathbf{f}(\mathbf{V}\mathbf{x}_r(t),\mathbf{u}(t))}_{\mathbf{r}(\mathbf{V}\mathbf{x}_r(t)),\mathbf{u}(t)} - \mathbf{r}(t)) = \mathbf{0}. \tag{3.9}$$

And enforcing the Petrov-Galerkin condition yields the desired ROM

$$\begin{aligned}\mathbf{E}_r\dot{\mathbf{x}}_r(t) &= \mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r(t),\mathbf{u}(t)), \qquad \mathbf{x}_r(0)=\mathbf{x}_{r,0}\\ \mathbf{y}_r(t) &= \mathbf{g}(\mathbf{V}\mathbf{x}_r(t)),\end{aligned} \tag{3.10}$$

where $\mathbf{E}_r = \mathbf{W}^T\mathbf{E}\mathbf{V}$ and $\mathbf{x}_{r,0} = (\mathbf{W}^T\mathbf{E}\mathbf{V})^{-1}\mathbf{W}^T\mathbf{E}\mathbf{x}(0)$.

3.3 Overview of nonlinear reduction methods

Since most practical, real-life dynamical systems are inherently nonlinear, this has led to many research efforts from the past decades to reduce large-scale nonlinear systems. Special classes of large-scale systems such as bilinear systems, DAE systems, and Hamiltonian systems have been successfully reduced, see, e.g., Lall et al. [177], Soberg et al. [272], Al-Baiyat et al. [3] and Fujimoto [121]. Polynomial approximation of weakly nonlinear systems has been carried by Chen [82], Rewienski and White [249, 250], and Benner [45]. Later on, the idea of reducing certain nonlinear systems by transforming them into quadratic bilinear form has also been proposed by Benner and Breiten [48] Gu [139], and Antoulas et al. [16]. Methods based on variational analysis were also proposed [47, 139, 232]. Besides this, Krylov subspace methods for special nonlinear system classes have also been studied in Refs. [62, 131].

3.3.1 Quadratic Method

Proposed by Chen [82], this method approximates the nonlinear function $\mathbf{f}(.)$ in (3.1) as a kronecker product formulation of $\mathbf{x}$, which is given as:

$$\begin{aligned}\mathbf{f}(\mathbf{x}(t),\mathbf{u}(t)) =& \mathbf{f}(\mathbf{0}) + \mathbf{A}_1\mathbf{x}(t) + \mathbf{A}_2(\mathbf{x}(t)\otimes\mathbf{x}(t)) \\ &+ \mathbf{A}_3(\mathbf{x}(t)\otimes\mathbf{x}(t)\otimes\mathbf{x}(t)) + ... + \mathbf{B}\mathbf{u}(t)\end{aligned} \tag{3.11}$$

where the matrices $\mathbf{A}_i \in \mathbb{R}^{n\times n}$ denote i-th partial derivative of $\mathbf{f}$ at $\mathbf{x} = \mathbf{0}$. The quadratic system thus obtained is given as:

$$\begin{aligned}\mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{f}(\mathbf{0}) + \mathbf{A}_1\mathbf{x}(t) + \mathbf{A}_2(\mathbf{x}(t)\otimes\mathbf{x}(t)) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{x}(t)).\end{aligned} \tag{3.12}$$

The projection matrix $\mathbf{V}$, in this method, is obtained as the basis of the *Krylov* subspace given as:

$$\text{colspan}(\mathbf{V}) = \mathcal{K}_r(\mathbf{A}_1^{-1}, \mathbf{A}_1^{-1}\mathbf{B}), \tag{3.13}$$

and the reduced quadratic model is obtained by using the one-sided approximation ansatz $\mathbf{x} \approx \mathbf{V}\mathbf{x}_r$ in (3.12) given as:

$$\begin{aligned}\mathbf{E}_r\dot{\mathbf{x}}_r(t) &= \mathbf{V}^T\mathbf{A}_1\mathbf{V}\mathbf{x}_r(t) + \mathbf{V}^T\mathbf{A}_2(\mathbf{V}\mathbf{x}_r(t)\otimes\mathbf{V}\mathbf{x}_r(t)) + \mathbf{V}^T\mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{V}\mathbf{x}_r(t)).\end{aligned} \tag{3.14}$$

It is assumed that $\mathbf{f}(\mathbf{0}) = \mathbf{0}$ otherwise is taken as a part of input signal.

3.3.2 Bilinearization Method

Based on Carleman linearization of nonlinear systems [261], the bilinear form was proposed by Bai et al. [33] and Phillips [231]. This involves the use of first two terms of the series in (3.11), so as to approximate the nonlinear function as:

$$\mathbf{f}(\mathbf{x}(t),\mathbf{u}(t)) \approx \mathbf{f}(0) + \mathbf{A_1}\mathbf{x}(t) + \mathbf{A_2}(\mathbf{x}(t)\otimes\mathbf{x}(t)) + \mathbf{B}\mathbf{u}(t). \tag{3.15}$$

As such the bilinear, input-affine system is obtained as follows (with $\mathbf{E} = \mathbf{I}$):

$$\dot{\mathbf{x}}_\otimes(t) = \mathbf{A}_\otimes\mathbf{x}_\otimes + \mathbf{N}_\otimes\mathbf{x}_\otimes\mathbf{u}(t) + \mathbf{B}_\otimes\mathbf{u}(t), \tag{3.16a}$$

$$\mathbf{y}(t) = \mathbf{C}_\otimes\mathbf{x}_\otimes, \tag{3.16b}$$

where

$$\begin{aligned}\mathbf{x}_\otimes = \begin{bmatrix}\mathbf{x}(t)\\ \mathbf{x}(t)\otimes\mathbf{x}(t)\end{bmatrix}, \quad \mathbf{A}_\otimes = \begin{bmatrix}\mathbf{A_1} & \mathbf{A_2}\\ 0 & \mathbf{A_1}\otimes\mathbf{I} + \mathbf{I}\otimes\mathbf{A_1}\end{bmatrix}, \\ \mathbf{N}_\otimes = \begin{bmatrix}0 & 0\\ \mathbf{B}\otimes\mathbf{I} + \mathbf{I}\otimes\mathbf{B} & 0\end{bmatrix}, \mathbf{B}_\otimes = \begin{bmatrix}\mathbf{B} & 0\end{bmatrix}^{\mathbf{T}}, \mathbf{C^T}_\otimes = \begin{bmatrix}\mathbf{C^T} & 0\end{bmatrix}^{\mathbf{T}}.\end{aligned} \tag{3.17}$$

After transforming the nonlinear system to bilinear form (3.16), there are several choices of calculating the projection basis. One particular choice was proposed by Philips and Joel [231]. It was shown that after constructing a series of Krylov subspaces for given $r_1, r_2, ..., r_K$, as follows:

$$\text{colspace}(\mathbf{V_1}) = \mathcal{K}_{r_1}(\mathbf{A}_{\otimes}^{-1}, \mathbf{B}_{\otimes}), \tag{3.18}$$

and

$$\text{colspace}(\mathbf{V}_k) = \mathcal{K}_{r_k}(\mathbf{A}_{\otimes}^{-1}, \mathbf{N}_{\otimes}\mathbf{V}_{k-1}), \tag{3.19}$$

for $1 < k < K$, the matrix $\mathbf{V}$ is calculated by taking the union of all the subspaces spanned by columns of $\mathbf{V}_k$, i.e.,

$$\text{colspan}(\mathbf{V}) = \bigcup_{k=1}^{K} \{\mathbf{V_k}\}. \tag{3.20}$$

Another choice of transformation matrix $\mathbf{V}$ was proposed by Bai and Skoogh [33], which matches as many multimoments as the reduced model, and is given as:

$$\text{colspace}(\mathbf{V_1}) = \mathcal{K}_{r_1}(\mathbf{A}_{\otimes}^{-1}, \mathbf{A}_{\otimes}^{-1}\mathbf{B}_{\otimes}), \tag{3.21}$$

and for $k > 1$,

$$\text{colspace}(\mathbf{V_k}) = \mathcal{K}_{r_k}(\mathbf{A}_{\otimes}^{-1}, \mathbf{A}_{\otimes}^{-1}\mathbf{N}_{\otimes}\mathbf{V_{k-1}}), \tag{3.22}$$

and the final transformation matrix $\mathbf{V}$ is constructed in a similar fashion, i.e.,

$$\text{colspan}(\mathbf{V}) = \bigcup_{k=1}^{K} \{\mathbf{V}_k\}. \tag{3.23}$$

After constructing the matrix $\mathbf{V}$, the reduced bilinear system is then obtained by using the approximation $\mathbf{x}_{\otimes} = \mathbf{V}\hat{\mathbf{x}}_{\otimes}$ in system (3.16) given as:

$$\dot{\hat{\mathbf{x}}}_{\otimes}(t) = \hat{\mathbf{A}}_{\otimes}\hat{\mathbf{x}}_{\otimes} + \hat{\mathbf{N}}_{\otimes}\hat{\mathbf{x}}_{\otimes}\mathbf{u}(t) + \hat{\mathbf{B}}_{\otimes}\mathbf{u}(t), \tag{3.24a}$$

$$\mathbf{y}(t) = \hat{\mathbf{C}}_{\otimes}\hat{\mathbf{x}}_{\otimes}, \tag{3.24b}$$

where

$$\mathbf{x}_{\otimes} = \mathbf{V}\hat{\mathbf{x}}_{\otimes}, \hat{\mathbf{A}}_{\otimes} = (\mathbf{V}^T\mathbf{A}_{\otimes}\mathbf{V}), \hat{\mathbf{N}}_{\otimes} = \mathbf{V}^T\mathbf{N}_{\otimes}\mathbf{V}, \hat{\mathbf{B}}_{\otimes} = \mathbf{V}^T\mathbf{B}_{\otimes} \text{ and } \hat{\mathbf{C}}_{\otimes} = \mathbf{C}\mathbf{V}_{\otimes}.$$

Another way of reducing bilinear systems is by using the Balanced Truncation (BT) method for bilinear system method which was originally developed by Achar and Nakhla [2], redeveloped by Condon and Ivanov [200], and also by Benner et al.[49]. This method is similar to standard BT method for linear systems. Although, this method outperforms moment-matching methods in terms of approximating quality, it involves solving the generalized *Lyapunov* equation which is computationally demanding.

3.3.3 Trajectory Piecewise Linear Approximation (TPWL)

TPWL method replaces the polynomial approximation of the nonlinear system with a piece-wise approximation. This method offers more robustness when dealing with strong nonlinearities. The idea is to linearize the nonlinear function $\mathbf{f}(.)$ at a number of linearization points $\mathbf{x}_i (i = 0, 1, ..., l)$ in response to some training input. The nonlinear function $\mathbf{f}(.)$ is then approximated by the weighted sum of these linear models i.e., $\mathbf{f}(\mathbf{x}(t)) \approx \sum_{i=0}^{l} w_i(\mathbf{f}(\mathbf{x}_i) + \mathbf{A}_i(\mathbf{x}(t) - \mathbf{x}_i))$, and the original system is represented as:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \sum_{i=0}^{l} w_i(\mathbf{x}(t))\mathbf{f}(\mathbf{x}_i) + \sum_{i=0}^{l} w_i(\mathbf{x}(t))\mathbf{A}_i(\mathbf{x}(t) - \mathbf{x}_i)) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{x}(t)), \end{aligned} \tag{3.25}$$

where $w_i, i = 0, 1, ..., l$ is a vector of weights and $\mathbf{A}_i$ is the Jacobian matrix of $\mathbf{f}(.)$ at $\mathbf{x}_i$. The projection matrix $\mathbf{V}$, in this case, is obtained as orthonormal basis of the following Krylov subspace which spans the reduced state-space.

$$\text{colspan}(\mathbf{V}) = \text{span}\{\mathbf{A}_0^{-1}\mathbf{B}, ..., \mathbf{A}_0^{-(r-1)}\mathbf{B}\}. \tag{3.26}$$

The reduction is then performed by using the approximation ansatz $\mathbf{x}(t) = \mathbf{V}\mathbf{x}_r(t)$ in (3.25).

$$\begin{aligned} \dot{\mathbf{x}}_r(t) &= \left(\sum_{i=0}^{l} \mathbf{V}^T\mathbf{A}_i\mathbf{V}\tilde{w}_i(\mathbf{x}_r(t)) \right) \mathbf{x}_r(t) \\ &\quad + \sum_{i=0}^{l} \mathbf{V}^T(\mathbf{f}(\mathbf{x}_i - \mathbf{A}_i\mathbf{x}_i)\tilde{w}_i(\mathbf{x}_r(t)) + \mathbf{V}^T\mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{g}(\mathbf{V}\mathbf{x}_r(t)). \end{aligned}$$

Here, the weights $\tilde{w}$ depend on $\mathbf{x_r}$ and can be computed using the information about the distances $\left\|\mathbf{x}_r - \mathbf{x}_{r_i}\right\|$ of the (projected) linearization points $\mathbf{V}^T\mathbf{x}_i$ from the current state $\mathbf{x}_r$. An error bound for TPWL was proposed in Ref. [250] alongwith some discussion on stability and passivity preservation. A down-side of this method, however, is the choice of an appropriate training input. The reduced model loses accuracy if the training inputs are chosen far away from the actual inputs. This happens because the computed trajectory departs from the actual behavior of the state vector $\mathbf{x}(t)$.

In continuation with this, a polynomial piece-wise approximation was proposed by Dong and Roychowdhury [93] instead of a linear piece-wise approximation for each piece. Similar ideas later emerged in various studies such as presented by Vasilyev et al. [297]. A more detailed review on piece-wise linearzation based reduction methods can be found in Ref. [151].

3.3.4 Proper Orthogonal Decomposition (POD)

One of the most successful nonlinear model order reduction method is the Proper Orthogonal Decomposition (POD) [210, 58] also known as Karhunen-Loéve Transform (KLT) or Principal Component Analysis (PCA). This method can be considered a general application of the SVD to the approximation of dynamical systems.

Let $\mathbf{x}(t_1), \mathbf{x}(t_2), ..., \mathbf{x}(t_{n_s})$ denote the snapshots of (3.1) taken at $t_j, j \in \{1, ..., n_s\}$ in the interval $[0, T]$. Suppose $\aleph$ represents the ensemble consisting the snapshots $\{\mathbf{x}(t_j)\}_{j=1}^{n_s}$ i.e.,

$$\aleph = \text{span}\{\mathbf{x}(t_1), ..., \mathbf{x}(t_{n_s})\}. \tag{3.27}$$

Then, POD aims at finding an orthonormal basis $\{\mathbf{v}_i\}_{i=1}^{r}$ for $\aleph$ that satisfies the minimization problem given as:

$$\min_{\{\mathbf{v}_i\}_{i=1}^{r}} \sum_{j=1}^{n_s} \left\| \mathbf{x}(t_j) - \sum_{i=1}^{r} \left\langle \mathbf{x}(t_j), \mathbf{v}_i \right\rangle \mathbf{v}_i \right\|_2^2, \tag{3.28}$$

subject to $\left\langle \mathbf{v}_i, \mathbf{v}_j \right\rangle = \delta_{ij}$. In practice, (3.28) is solved by taking the SVD of the snapshot matrix $\mathbf{X} \in \mathbb{R}^{n \times n_s}$ (whose columns are the time displaced solutions of the system) i.e, $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$, and first $r << \min(n, n_s)$ columns of $\mathbf{U}$ are then utilized to construct the orthonormal project matrix $\mathbf{V}_{\text{POD}}$. The procedure is summed up in Algorithm 2.

Algorithm 2 Proper Orthogonal Decomposition (POD)

Input: Snapshot matrix $\mathbf{X} = [\mathbf{x}(t_1)\ \mathbf{x}(t_2)...\ \mathbf{x}(t_{n_s})]$, tolerance parameter ϵ.

Output: POD basis: $\mathbf{V}_{\text{POD}}$

1: $[\mathbf{U}, \mathbf{\Sigma}, \mathbf{V^T}] = svd(\mathbf{X})$, $\mathbf{\Sigma}$ =diag$(\sigma_1, \sigma_2, ..., \sigma_{r_x})$.

2: $\mathbf{V}_{\text{POD}} = \mathbf{U}(:, 1:r)$, r s.t., $\sum_{i=r+1}^{r_x} \sigma_i / \sum_{i=1}^{r_x} \sigma_i < \epsilon$, r_x is the number of non-zero singular values of matrix $\mathbf{\Sigma}$.

Note that, for the most MOR settings, the dimensions of the snapshot matrix $\mathbf{X} \in \mathbb{R}^{n \times n_s}$ with $n > n_s$, i.e., the number of dimension of system is much greater that than the number of snapshots, resulting in a "tall-and-skinny" matrix $\mathbf{X}$. Therefore, it is preferred to use a economy version of the SVD resulting in the dimension of $\mathbf{U} \in \mathbb{R}^{n \times n_s}, \mathbf{\Sigma} \in \mathbb{R}^{n_s \times n_s}$, and $\mathbf{V} \in \mathbb{R}^{k \times n_s}$ (cf. Fig. 2.2)

Given its conceptual simplicity, POD is a straightforward data-driven approach and performs satisfactorily well for a number of nonlinear systems. Also, the reduced model of size r can be selected according to the decay of singular values for a given tolerance ϵ. However, there are some drawbacks associated with this method. Firstly, the low-rank approximation via POD relies on expensive measurements of the full-sized model for some training inputs $\mathbf{u}_{\text{train}}(t)$. This leads to substantial offline computational costs, especially when the FOM is repeatedly

solved for varying design parameters, initial or boundary conditions. Secondly, the quality of ROM depends on the training inputs used while obtaining the snapshot ensemble. If the latter doesn't capture the system's dynamics at all operating points, then the performance of ROM degrades for scenarios for which it was not trained.

Besides the classical POD technique, there exist several other methods which are related to the POD method. Examples include balanced-POD proposed by Willcox and Peraire [306], Kunisch and Volkwein [174], Hinze and Volkwein [150], Kunisch and Volkwein [175], Grepl et al. [135] and Astrid et al. [28], Reduced Basis methods by Quarteroni et al. [236, 149], empirical Gramians [179], the *shifted* POD (sPOD) method by Reiss et al.[248] for advection-dominated PDEs, and methods based on deep/manifold learning [227, 285, 183]. Also, POD has been widely employed within the spatial least-square Petrov-Galerkin (LSPG) framework [69, 70], and the recently developed space-time LSPG method [85] involving the higher-order singular value decomposition (HOSVD).

3.3.5 Nonlinear balancing

The concept of balancing for nonlinear systems was first proposed by Scherpen [267] which is an extension of balancing for linear systems in the sense that it is based on input-output energy of the nonlinear system. This involves solving two Hamilton-Jacobi-Bellman (HJB) PDEs representing nonlinear counterparts of linear Lyapunov equations (2.40). However, for problems of large-scale nature, the solution of HJB PDEs becomes highly expensive due to the *curse of dimensionality*. Hence, state-dependent Lyapunov equations have been developed to avoid this issue. These include *differential balancing* based on variational systems and contraction theory by Kawano et al. [164], and *dynamic balancing* based on dynamic controllability and observability extensions presented by Sassano et al. [259, 260]. This further encouraged the development of energy-based methods and the notion of Hankel operator, see e.g., Gray and Mesko [133], Scherpen and Gray [266], Scherpen and Van-der-Schaft [268] and Fujimoto and Scherpen [122, 123].

3.4 Moment-matching in nonlinear systems

The relation between moments and Sylvester equation was first proposed by Gallivan et al. [126, 127]. Based on this, the notion of nonlinear moments was firstly presented by Astolfi [23]. Since then, methods on linear moment matching and, especially, nonlinear moment-matching (NLMM) have been developed in many publications [24, 25, 26, 27, 29]. A class of (nonlinear) parameterized ROMs achieving moment matching was defined in Ref. [154]. A two-sided, nonlinear moment matching theory was developed by Ionescu and Astolfi [155, 156]. The problem of MOR via moment-matching for linear and nonlinear differential time-delay systems with discrete and distributed-delays was studied by Scarciotti

and Astolfi [263]. In addition to this, the online estimation of moments of linear and nonlinear systems from input/output measurements has been proposed in Ref. [264, 265]. The study presented new algorithms to construct ROM's that asymptotically match the moments of an unknown nonlinear system to be reduced by solving a recursive, moving window, least-square estimation problem using input/output snapshot measurements. In reference to this, Maria et al. [295] developed a practical simulation-free, NLMM algorithm in which certain numerical simplifications were proposed that avoid the expensive solution of Sylvester PDE. Recently, a nonlinear MOR framework based on NLMM with dynamic mode decomposition (DMD) has been proposed in Ref. [241]. The framework involves the computation of the offline projection matrix $\mathbf{V}$ by the application of NLMM, whereas the underlying nonlinearity is approximated via DMD modes. Also, a parametric nonlinear MOR method based on NLMM has been proposed in Ref. [240](see also [239, 237, 238, 242]). Faedo et al. [104] recently extended this idea to wave energy systems. Furthermore, an optimal $\mathcal{H}_2$-norm based MOR method using time-domain moment-matching was proposed by Ion Necoara and Ionescu [216].

Similar to the notion of moments defined for linear systems in Section 2.2.7, we describe the nonlinear enhancement of moment-matching for large-scale systems. The idea is based on the concepts emerging from output regulation of nonlinear systems [173], the center manifold theory [72], and the steady-state response of nonlinear-systems [157, 67, 158]. We refer the reader to Ref. [243] for an excellent review.

Consider the following nonlinear exogenous (signal-generator) system:

$$\dot{\boldsymbol{\zeta}}(t) = \boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) \qquad \boldsymbol{\zeta}(0), = \boldsymbol{\zeta}_0 \neq 0, \tag{3.29a}$$

$$\mathbf{u}(t) = \boldsymbol{\varrho}(\boldsymbol{\zeta}(t)), \tag{3.29b}$$

where $\boldsymbol{\zeta} \in \mathbb{R}^r, \boldsymbol{\varpi}(\boldsymbol{\zeta}) \in \mathbb{R}^r \to \mathbb{R}^r$ and $\boldsymbol{\varrho}(\boldsymbol{\zeta}) \in \mathbb{R}^r \to \mathbb{R}^m$ are smooth mapping such that $\boldsymbol{\varpi}(\mathbf{0}) = \mathbf{0}$ and $\boldsymbol{\varrho}(\mathbf{0}) = \mathbf{0}$. It is hereby assumed that the exogenous system (3.29) is observable $(\boldsymbol{\varrho}, \boldsymbol{\varpi}, \boldsymbol{\zeta_0})$, i.e., the output trajectories corresponding to any initial conditions do not coincide. Also, its is assumed that the point $\boldsymbol{\zeta_0}$ is a stable equilibrium (in ordinary sense of Lyapunov) such that inputs generated by such a system remain bounded. Furthermore, the input signal $\mathbf{u}(t)$ is assumed to be persistently exciting in time [157]. This implies that every point $\boldsymbol{\zeta^o}$ is Poisson stable, such that no trajectory can decay to zero as $t \to \infty$. When both the above conditions are met, we define the signal generator to be "neutrally stable" [157] i.e., if the output signal of such a signal generator is fed to the input of nonlinear system (3.1), the steady-state of output of such an interconnection is guaranteed to be well-defined. The signal generator thus essentially captures the requirement that one is interested in studying the the behavior of system (3.1) only in specific circumstances. The interconnection of exogenous system (3.29) with nonlinear

system (3.1) is given as:

$$\mathbf{E}\frac{d\mathbf{x}(t)}{dt} = \mathbf{f}(\mathbf{x}(t), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t)), \tag{3.30a}$$

$$\mathbf{y}(t) = \mathbf{g}(\mathbf{x}(t)). \tag{3.30b}$$

The steady-state of such an interconnection is given as:

$$\mathbf{y}(t) = \underbrace{\mathbf{g}\boldsymbol{\tau}(\mathbf{x}_0 - \boldsymbol{\wp}(\boldsymbol{\zeta_0}))}_{\mathbf{y_p}(t)} + \underbrace{\mathbf{g}\boldsymbol{\wp}(\boldsymbol{\zeta}(t))}_{\mathbf{y_h}(t)}, \tag{3.31}$$

where $\mathbf{y_p}(t)$ is the transient component of the solution with *nonlinear transient mapping* given by $\boldsymbol{\tau}(\mathbf{x}_0)$, and $\mathbf{y_h}(t)$ corresponds to the steady-state component of the solution where the mapping $\boldsymbol{\wp}(\boldsymbol{\zeta})$ is a local mapping defined in the neighborhood of $\boldsymbol{\zeta} = 0$. Since the nonlinear system (3.1) is assumed to be exponentially stable, therefore the transient solution decays exponentially, i.e., $\lim_{t\to\infty} \mathbf{y_h}(t) = 0$, and $\mathbf{y_{ss}}(t) = \mathbf{g}(\boldsymbol{\wp}(\boldsymbol{\zeta}(t)))$ is the steady-state response of the interconnected system. Thus, interconnecting the nonlinear system (3.1) with the signal generator (3.29) corresponds to exciting the nonlinear system with user-defined inputs. Now, to provide a generalized notion of moments for nonlinear systems, the following assumption is required (as provided in Ref.[26])

Assumption: The unique mapping $\boldsymbol{\wp}(\boldsymbol{\zeta})$ solves the nonlinear *Sylvester* equation given as:

$$\mathbf{E}\frac{\partial \boldsymbol{\wp}(\boldsymbol{\zeta}(t))}{\partial \boldsymbol{\zeta}(t)}\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) = \mathbf{f}(\boldsymbol{\wp}(\boldsymbol{\zeta}(t)), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t))). \tag{3.32}$$

By the center manifold theory, the interconnected system (3.30) possesses a locally well-defined invariant manifold at $(\mathbf{x_{eq}}, \boldsymbol{\zeta_{eq}} = (\mathbf{0}, \mathbf{0}))$, given as $\mathcal{M} = \{(\mathbf{x}, \boldsymbol{\zeta}) \in \mathbb{R}^{n+r} : \mathbf{x} = \boldsymbol{\wp}(\boldsymbol{\zeta}(t))\}$ where $\boldsymbol{\wp}(\boldsymbol{\zeta}(t))$ solves (3.32).

Definition 3.1 (Steady-state notion of nonlinear moments) *[25, 26]* The moment of system (3.1) at $\boldsymbol{\varpi}(\boldsymbol{\zeta})$ is defined by function $\mathbf{g}(\boldsymbol{\wp}(\boldsymbol{\zeta}))$ under the aforesaid assumptions. Furthermore, the 0-th nonlinear moment of system (3.1) at $(\boldsymbol{\varpi}(\boldsymbol{\zeta}(t))), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t)), \boldsymbol{\zeta_0})$ is related to the (locally well-defined) steady-state response $\mathbf{y_{ss}}(t) = \mathbf{g}(\boldsymbol{\wp}(\boldsymbol{\zeta}(t)))$ where the mapping $\boldsymbol{\wp}(\boldsymbol{\zeta})$ solves the nonlinear Sylvester equation (3.32). ▲

Based on Definition 3.1, the notion of nonlinear moment-matching in terms of steady-state response matching can be formulated as follows:

Theorem 3.1 (Nonlinear moment-matching) *[25, 26] The moments of the nonlinear system (3.1) at* $(\boldsymbol{\varpi}(\boldsymbol{\zeta}), \boldsymbol{\varrho}(\boldsymbol{\zeta}), \boldsymbol{\zeta_0})$ *coincide with the (well-defined) steady-state response of the output of the interconnected system (3.30) when the aforesaid assumptions hold.* ▲

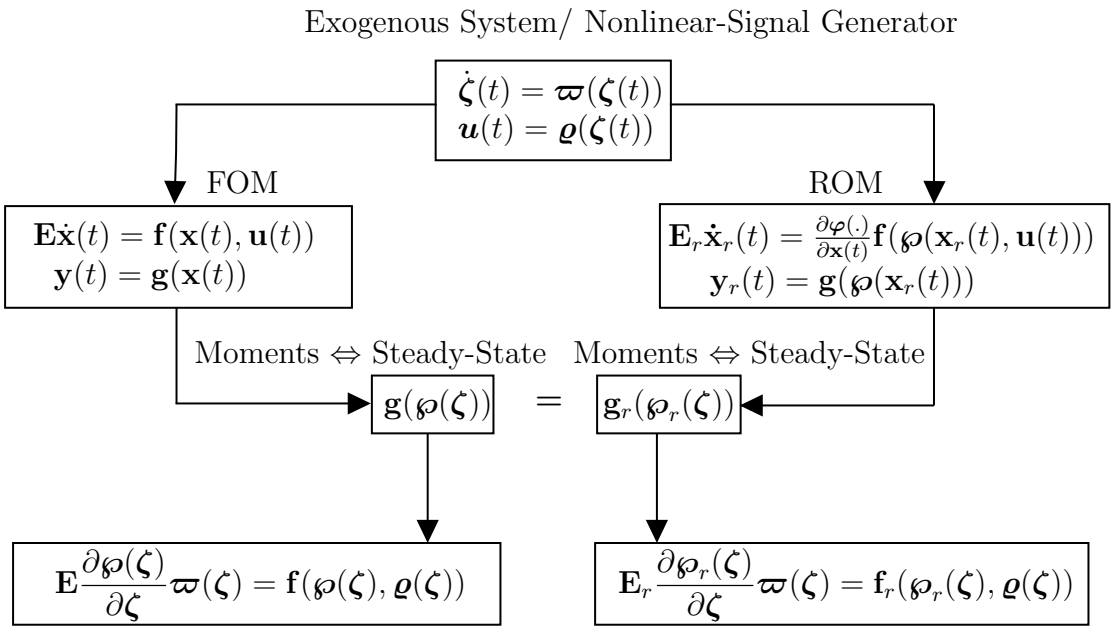

Figure 3.1: Time-domain illustration of nonlinear moment-matching in terms of steady-state response matching between FOM and ROM

Thus, similar to linear systems, a nonlinear reduced model can be defined that achieves moment-matching with original system (3.1) at $(\boldsymbol{\varpi}(\boldsymbol{\zeta}), \boldsymbol{\varrho}(\boldsymbol{\zeta}), \boldsymbol{\zeta_0})$.

Definition 3.2 (Nonlinear ROM via moment-matching) *[25, 26]* A reduced-order model of the form (3.8), given as:

$$\begin{aligned} \mathbf{E}_r\dot{\mathbf{x}}_r(t) &= \left.\frac{\partial \boldsymbol{\varphi}(\boldsymbol{\omega}(\mathbf{x}(t), \mathbf{u}(t)))}{\partial \mathbf{x}(t)}\right|_{\mathbf{x}(t)=\boldsymbol{\wp}(\mathbf{x}_r(t))} \mathbf{f}(\boldsymbol{\wp}(\mathbf{x}_r(t), \mathbf{u}(t))), \quad \mathbf{x}_r(0) = \mathbf{x}_{r,0}, \\ \mathbf{y}_r(t) &= \mathbf{g}(\boldsymbol{\wp}(\mathbf{x}_r(t))), \end{aligned} \tag{3.33}$$

of dimension r ($r << n$) matches moments with the full-order system (3.1) at $(\boldsymbol{\varpi}(\boldsymbol{\zeta}), \boldsymbol{\varrho}(\boldsymbol{\zeta}), \boldsymbol{\zeta_0})$, if the reduced nonlinear Sylvester equation given as:

$$\mathbf{E}_r\frac{\partial \boldsymbol{\wp}_r(\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}}\boldsymbol{\varpi}(\boldsymbol{\zeta}) = \mathbf{f}_r(\boldsymbol{\wp}_r(\boldsymbol{\zeta}), \boldsymbol{\varrho}(\boldsymbol{\zeta})), \tag{3.34}$$

is uniquely solved by $(\mathbf{g}_r(\boldsymbol{\zeta}))$ such that

$$\mathbf{g}(\boldsymbol{\wp}(\boldsymbol{\zeta})) = \mathbf{g}_r(\boldsymbol{\wp}_r(\boldsymbol{\zeta})). \tag{3.35}$$

▲

Thus, moment-matching for nonlinear systems can be interpreted as the exact matching of the steady-state response of the FOM and ROM (cf. Fig. 3.1) when

both systems are excited from the inputs from non-linear signal generator (3.29). For other inputs, the steady state is interpolated. Note here that the *Sylvester* equation unlike in the linear case, is state dependent, non-linear partial differential equation of dimension $(n \times 1)$.

The projection-based technique to obtain a reduced-order model achieving moment-matching is given by (3.8) with $\wp(\mathbf{x}_r(t))$ as the solution Sylvester PDE (3.32). However, a non-projective approach consists of parametrizing the family of ROMs achieving moment-matching with FOM (3.1) at $(\boldsymbol{\varpi}(\boldsymbol{\zeta}), \boldsymbol{\varrho}(\boldsymbol{\zeta}), \boldsymbol{\zeta_0})$ w.r.t the reduced input matrix function $\boldsymbol{\delta}(\mathbf{x}_r(t)) : \mathbb{R}^r \rightarrow \mathbb{R}^{r\times m}$ yielding a ROM in *input-affine* representation as follows:

$$\dot{\mathbf{x}}_r(t) = \boldsymbol{\varpi}(\mathbf{x}_r(t)) - \boldsymbol{\delta}(\mathbf{x}_r(t))\boldsymbol{\varrho}\mathbf{x}_r(t) + \boldsymbol{\delta}(\mathbf{x}_r(t))\mathbf{u}(t), \tag{3.36a}$$

$$\mathbf{y}_r(t) = \mathbf{g}(\wp(\mathbf{x}_r(t))), \tag{3.36b}$$

where the free mapping $\boldsymbol{\delta}(\mathbf{x}_r)$ can be used to impose e.g. asymptotic stability, passivity, a prescribed relative degree and zero dynamics.

3.5 Hyper-reduction in nonlinear systems

The nonlinear reduction methods reduce the number of equation from n to r yielding the ROM structure (3.8) or (3.10) depending upon the type of projection technique used. This results in a drastic reduction in the computational cost associated with solving the linear system of equations via any time-integration scheme. However, the complexity of evaluating the nonlinear term $\mathbf{f}(\mathbf{x}_r, \mathbf{u})$ still remains as that of the original problem. For this, consider the nonlinear term of the ROM (3.8) :

$$\underbrace{\mathbf{W}^T}_{r\times n}\underbrace{\mathbf{f}(\mathbf{V}\mathbf{x}_r(t))}_{n\times 1} = \left\langle \mathbf{f}(\mathbf{V}\mathbf{x}_r(t)), \mathbf{V} \right\rangle. \tag{3.37}$$

This inner product is solved by first expanding the variable $\mathbf{x}_r(t) \in \mathbb{R}^r$ to n-dimensional vector $\mathbf{V}\mathbf{x}_r(t) \in \mathbb{R}^n$ which requires $2nr$ flops. Then, the nonlinearity $\mathbf{f}(\mathbf{V}\mathbf{x}_r)$ is evaluated. Finally, the projection with the reduction basis $\mathbf{W}$ is solved to obtain $\mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r(t))$. Thus, $4nr$ flops are required as a result of two matrix-vector products required to form the argument of $\mathbf{f}$, and then to form the projection. As a result, solving the reduced system might still be costly as solving the full system.

Similar inefficiency occurs when solving the reduced model (3.8) for steady nonlinear PDEs by Newton iteration. During each iteration, the Jacobian matrix $\mathbf{J}(\mathbf{x}) = \partial\mathbf{f}(\mathbf{x})/\partial\mathbf{x}$ is solved besides the nonlinear function $\mathbf{f}$, and the cost of computation still depends on full-dimension n:

$$\mathbf{J}_r(\mathbf{x}_r) = \underbrace{\mathbf{W}^T}_{r\times n}\underbrace{\mathbf{J}(\mathbf{V}\mathbf{x}_r)}_{n\times n}\underbrace{\mathbf{V}}_{n\times r} \tag{3.38}$$

i.e., the Jacobian function is first expanded in the high-dimension settings, and then projected to obtain the reduced form. The computational complexity for

computing the Jacobian is roughly $\mathcal{O}(2n^2r+2nr^2+2nr)$ for a dense $\mathbf{J}$ and $2n^2r$ for a sparse $\mathbf{J}$. Hence, these numerical complexities renders the dimension reduction techniques inefficient.

To address these challenges, various *hyper-reduction* scheme have been proposed. The idea is to find a low-rank representation of the nonlinear function and its Jacobian for an efficient reduction of the overall problem. These methods are usually employed after the dimensionality reduction of state-space have been performed. Next, we discuss some of the prominent hyper-reduction methods:

3.5.1 Polynomial system representation

One possibility is to express the nonlinear function analytically through polynomial tensors to avoid evaluating the nonlinear terms on the element level within the FE code. This is achieved by representing the nonlinear function $\mathbf{f}(\mathbf{x}, \mathbf{u})$ as a polynomial system using a Taylor series expansion (3.11) or a quadratic form (3.12). Once a polynomial system representation is obtained, the dimensional reduction can be carried out using either a linear or nonlinear projection. Although the full-order tensors are primarily sparse, their storage becomes prohibitive for large n. This is especially true if the nonlinear function is represented by higher-order polynomials like cubic or quartic. To illustrate the idea, consider a cubic representation of a nonlinear function $\mathbf{f}$ as follows:

$$\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{B}(\mathbf{x} \otimes \mathbf{x}) + \mathbf{C}(\mathbf{x} \otimes \mathbf{x} \otimes \mathbf{x}). \tag{3.39}$$

One way to obtain the reduced approximation of $\mathbf{f}$ is by using linear bases $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n\times r}$ for projection. The reduced model is then given as:

$$\mathbf{f}_r(\mathbf{x}_r) = \mathbf{A}_r\mathbf{x}_r + \mathbf{B}_r(\mathbf{x}_r \otimes \mathbf{x}_r) + \mathbf{C}_r(\mathbf{x}_r \otimes \mathbf{x}_r \otimes \mathbf{x}_r), \tag{3.40}$$

where the reduced matrices

$$\mathbf{A}_r = \mathbf{W}^T\mathbf{A}\mathbf{V}, \mathbf{B}_r = \mathbf{W}^T\mathbf{B}(\mathbf{V} \otimes \mathbf{V}), \text{ and } \mathbf{C}_r = \mathbf{W}^T\mathbf{C}(\mathbf{V} \otimes \mathbf{V} \otimes \mathbf{V}),$$

of dimensions $\mathbf{A} \in \mathbb{R}^{r\times r}, \mathbf{B}_r \in \mathbb{R}^{r\times r^2}$, and $\mathbf{C} \in \mathbb{R}^{r\times r^3}$. The smaller costs of evaluating the reduced polynomial function $\mathbf{f}_r$ gives considerable saving than solving $\mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r)$.

3.5.2 Piecewise linear approximation

Another possibility is to represent the nonlinear function by a piecewise linear approximation (PWL), as follows:

$$\mathbf{f}_r(\mathbf{x}_r) = \mathbf{W}^T \left(\sum_{i=0}^{l} w_i(\mathbf{x}_r(t))\mathbf{f}(\mathbf{x}_i) + \sum_{i=0}^{l} w_i(\mathbf{x}_r(t))\mathbf{A}_i(\mathbf{V}\mathbf{x}(t) - \mathbf{x}_i)) \right). \tag{3.41}$$

The nonlinearity is approximated as a weighted interpolation of reduced linearized matrices, obtained via linearizations at different given points $\{\mathbf{x}_i\}_{i=0}^{l}$.

3.5.3 Classical hyper-reduction methods

Unlike the above mentioned methods, the classical hyper-reduction methods like the Discrete Empirical Interpolation (DEIM) [80] by Chaturantabut and Sorensen or the Energy-Conserving Sampling and Weighting (ECSW) [106] method by Farhat et al. employ a different approach. The idea is to compute the nonlinear function only at certain grid points of the mesh.

3.5.4 Discrete Empirical Interpolation Method (DEIM)

The core ideas of DEIM emerge from empirical interpolation method(EIM) [36] by Barrault et al., Gappy POD [305, 103], and the missing point estimation method [28] by Astrid et al.

In DEIM, the non-linear function $\mathbf{f}(\mathbf{x})$ is projecting onto an approximate subspace spanned by a basis of dimension $m_d << n$, i.e.,

$$\mathbf{f}(\tau) \approx \mathbf{U_d}\mathbf{c}(\tau), \tag{3.42}$$

where for nonlinear time dependent PDEs $\tau = t$, $\mathbf{U_d} = [\mathbf{u}_1, ..., \mathbf{u}_{m_d}] \in \mathbb{R}^{n\times m_d}$, and $\mathbf{c}(\tau)$ as the coefficient vector. Next, consider a boolean matrix $\mathbf{P}$ with:

$$\mathbf{P} = [\mathbf{e}_{p_1}, ..., \mathbf{e}_{p_{m_d}}] \in \mathbb{R}^{n\times m_d}, \tag{3.43}$$

where $\mathbf{e}_{pi} = [0, ..., 0, 1, 0, ..., 0]^T \in \mathbb{R}^n$ is the p_ith column of the identity matrix $\mathbf{I}_n \in \mathbb{R}^{n\times n}$ for $i = 1, .., m_d$, then

$$\mathbf{P^T f}(\tau) = (\mathbf{P^T U_d})\mathbf{c}(\tau), \tag{3.44}$$

and approximation (3.42) becomes:

$$\mathbf{f}(\tau) \approx \mathbf{U_d}\mathbf{c}(\tau) = \mathbf{U_d}(\mathbf{P^T U_d})^{-1}\mathbf{P^T f}(\tau), \tag{3.45}$$

where $\mathbf{P^T U_d}$ is nonsingular. The DEIM algorithm to obtain $\vec{p} = [p_1, ..., p_{m_d}]^T$ is as follows:

Algorithm 3 Discrete Empirical Interpolation Method (DEIM)[80]

INPUT: $\{\mathbf{u}_l\}_{l=1}^{m_d} \subset \mathbb{R}^n$ linearly independent
OUTPUT: $\vec{p} = [p_1, ..., p_{m_d}]^T$

1: $[|\rho|, p_1] = max\{|\mathbf{u}_1|\}$.
2: $\mathbf{U_d} = [\mathbf{u}_1],\ \mathbf{P} = [\mathbf{e}_{p_1}],\ \vec{p} = [p_1]$.
3: **for** $l = 2$ to m_d **do**
4: Solve $(\mathbf{P}^T\mathbf{U_d})\mathbf{c} = \mathbf{P}^T\mathbf{u}_l$ for $\mathbf{c}$
5: $\mathbf{r} = \mathbf{u}_l - \mathbf{U_d}\mathbf{c}$
6: $[|\rho|, p_l] = max\{|\mathbf{r}|\}$
7: $\mathbf{U_d} \leftarrow [\mathbf{U_d}\ \ \mathbf{u}_l],\ \mathbf{P} \leftarrow [\mathbf{P}\ \ \mathbf{e}_{p_l}],\ \vec{p} \leftarrow \begin{bmatrix}\vec{p}\\ p_l\end{bmatrix}$.
8: **end for**

From the DEIM Algorithm, a set of indices are constructed inductively on the input basis. The input basis matrix $\mathbf{u}_l$ is obtained by sampling the nonlinear function $\mathbf{f}$ at certain time-instances $t = 1, 2, ..., t_{n_s}$. Thus, similar to the snapshot matrix $\mathbf{X} = [\mathbf{x}(t_1)\ \mathbf{x}(t_2)...\ \mathbf{x}(t_{n_s})] \in \mathbb{R}^{n \times n_s}$ obtained for POD, the *nonlinear function snapshots* $\mathbf{F} = [\mathbf{f}(\mathbf{x}(t_1))\ \mathbf{f}(\mathbf{x}(t_2))...\ \mathbf{f}(\mathbf{x}(t_{n_s}))] \in \mathbb{R}^{n \times n_s}$ are gathered from the FOM for some training signal $\mathbf{u}_{\text{train}}(t)$. After that, the SVD of the matrix $\mathbf{F}$ yields the columns of matrix $\mathbf{u}_l$, where the order of the input basis m_d is chosen according to the dominant singular values of the nonlinear function snapshot matrix. The DEIM approximation of the nonlinear term $\mathbf{f}()$ then reads:

$$\mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r(t)) = \mathbf{W}^T\mathbf{U}_\mathbf{d}(\mathbf{P}^T\mathbf{U}_\mathbf{d})^{-1}\mathbf{P}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r(t)), \tag{3.46}$$

and in case the nonlinear function $\mathbf{f}$ is evaluated component-wise at its input vector, we can write (3.46) as:

$$\mathbf{W}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r(t)) = \underbrace{\mathbf{W}^T\mathbf{U}_\mathbf{d}(\mathbf{P}^T\mathbf{U}_\mathbf{d})^{-1}}_{\text{precomputed: } r\times m_d}\underbrace{\mathbf{f}(\mathbf{P}^T\mathbf{V}\mathbf{x}_r(t))}_{m_d\times 1} \tag{3.47}$$

Since the term $\mathbf{W}^T\mathbf{U}_\mathbf{d}(\mathbf{P}^T\mathbf{U})^{-1}$ is independent of t, it can be precomputed by solving the system of ODEs. Furthermore, $\mathbf{P}^T\mathbf{V}\mathbf{x}_r(t) \in \mathbb{R}^{m_d}$ can be obtained by extracting the rows $p_1, p_2, ..., p_{m_d}$ of $\mathbf{V}$, and then multiplying against $\mathbf{x}_r$ which incurs $2rm_d$ cost. Here the indices $\{p_1, p_2, ..., p_{m_d}\}$ are obtained via Algorithm (3). Thus, the overall cost of evaluating the nonlinear term $\mathbf{f}$, via DEIM, becomes roughly $\mathcal{O}(4rm_d)$ which is is independent of n.

Similarly the reduced Jacobian $\mathbf{J}_r(\mathbf{x}_r)$ can be approximated via DEIM as:

$$\mathbf{J}_r(\mathbf{x}_r) = \underbrace{\mathbf{W}^T\mathbf{U}_\mathbf{d}(\mathbf{P}^T\mathbf{U}_\mathbf{d})^{-1}}_{\text{precomputed: } r\times m_d}\underbrace{\mathbf{J}(\mathbf{P}^T\mathbf{V}\mathbf{x}_r(t))}_{m_d\times m_d}\underbrace{\mathbf{P}^T\mathbf{V}}_{m_d\times r} \tag{3.48}$$

DEIM is a straightforward approach and can be applied immediately after any projection based model reduction technique. An error bound of the approximation is also available [80] which is given as:

$$\left\|\mathbf{f} - \mathbf{f}_{\text{DEIM}}\right\| \leq \left\|(\mathbf{P}^T\mathbf{U}_\mathbf{d})^{-1}\right\|_2 \mathcal{E}_*(\mathbf{f}), \tag{3.49}$$

where $\mathcal{E}_*(\mathbf{f}) = \left\|\right\|_2$ is the minimum 2-norm error of $\mathbf{f}$. There is good motivation of using DEIM especially with POD, as the time snapshots required for the DEIM are already obtaining while training POD. Thus, POD-DEIM is considered as a benchmark nonlinear MOR reduction technique for reducing both the state as well as the nonlinear function.

Besides the classical DEIM approach, there also exists different variants and extension of this method. For instance, the QDEIM [94] by Drmac and Gugercin is even more efficient that the classical DEIM approach. Similarly, the unassembled DEIM (UDEIM) [286] technique proposed by Paolo and Daniel extends the regular DEIM to FE applications. A symmetrized version of DEIM has been proposed for

nonlinear port-Hamiltonian systems [79]. The matrix version of DEIM (MDEIM) [308] proposed by Wirtz et al. for matrix-DEIM approximation of nonlinearity Jacobians.

Furthermore, the Gauss-Newton with Approximated Tensors (GNAT) method [70] by Carlberg et al. is another famous DEIM-related hyper-reduction technique which has also been recently extended to structural dynamics [71].

3.6 Issues in nonlinear MOR

Although nonlinear MOR methods have seen a prominent success, there are still some unresolved queries/issues, which are enlisted as follows:

1. A common issue among all nonlinear reduction methods is the lack of practical global error bounds for the reduced-order model.

2. Nonlinear moment-matching based MOR method presented in Refs.[24, 25, 26, 27] involves the a priori solution of the nonlinear, state-dependent, Sylvester-like PDE (3.32) which is computationally expensive.

3. There is no apriori knowledge of acceptable training inputs and linearization points in the TPWL method.

4. An optimal choice for the number of multi-moments in the bilinearization methods is not known apriori.

3.7 Chapter summary

In summary, we have discussed the model order reduction problem in nonlinear dynamical systems. We have explained the use of linear and nonlinear Petro-Galerkin projection frameworks in nonlinear systems. Then, we presented an overview of some of the most widely used MOR approaches for nonlinear models. We have also described the moment-matching in nonlinear systems, which is an extension of the linear systems. Finally, we have addressed the issue of reducing the nonlinear function, and in this regard, discussed various hyper-reduction procedures.

Chapter 4

Efficient Frameworks for Nonlinear Model Order Reduction

In the previous chapter, we explained the nonlinear model order reduction via linear and nonlinear projection methods. However, we can also classify the reduction methods based on their fundamental methodology. The first group is the *simulation-based* methods. These methods rely on taking the expensive measurements of the full order model to form the reduced manifold. Examples include Proper Orthogonal Decomposition (POD), Trajectory Piecewise Linear Method (TPWL), empirical Gramians, balanced POD, and the Reduced Basis (RB) method.

On the other hand, there exist another group of methods that are based on *simulation-free* approaches. These methods extract the most dominant nonlinear dynamics by system-theoretic measures rather than from simulated data. These include modal derivatives, nonlinear balanced truncation, Krylov subspace methods for polynomial systems, and the nonlinear moment-matching method. The latter method represents a promising approach from a system-theoretic perspective, as it avoids the expensive measurements of the full system to obtain the reduced model. Thus, in the following, we will focus on this method for reducing general nonlinear systems.

In this chapter, we will use some numerical simplifications for the underlying Sylvester nonlinear PDE to arrive at a system of nonlinear algebraic equations in a step-by-step manner. Then, we will propose some efficient reduction frameworks for nonlinear systems. These are (i) Nonlinear Moment-Matching (NLMM) with Discrete Empirical Interpolation Method (DEIM), (ii) Nonlinear Moment-Matching (NLMM) with Dynamic Mode Decomposition (DMD), and (iii) Second-Order Nonlinear Moment-Matching with DEIM (SONLMM-DEIM) (discussed in Chapter 5). We will extensively discuss each reduction framework and provide numerical algorithms for their implementation. Finally, we will illustrate the application of these algorithms on different nonlinear systems and present a qualitative and quantitative comparison of reduced models obtained with POD-DEIM.

4.1 Approximated nonlinear moment matching

The nonlinear model reduction via moment-matching described in Section 3.4 requires the solution of the nonlinear, state-dependent Sylvester PDE (3.32) which is computationally demanding. It's solution either requires symbolic computations or numerical solutions. The latter involves solving a system of nonlinear ODEs after the reduced spatial discretization of the PDE. As we intend to reduce large-scale nonlinear systems, only numerical methods come into consideration, which ideally should avoid any costly simulation. As such, certain numerical simplifications are proposed by Maria et al. [296, 295] to reduce the computational costs involved in solving the Sylvester PDE. These simplifications are motivated by POD method, which uses a linear projection and time-snapshots to reduce the nonlinear system. The result is that the nonlinear PDE is converted into a system of nonlinear algebraic equations, and a numerically feasible algorithm is obtained for the dimension reduction. Note that these simplification are related to asymptotic expansion (Poincaré/naive expansion) [213, 298, 162] or the variational approach [252] for approximating the solution of nonlinear differential equations.

4.1.1 Simplifications

The numerical simplifications to solve the Sylvester PDE (3.32)are three-fold (i) using a linear projection, (ii) to consider the column-wise operation of the equation, and (iii) to collect time-snapshots for discrete samples in time. Now, depending on the type of signal generator involved, we distinguish between the following cases:

Nonlinear Signal Generator

1. **Linear projection:** Instead of using a non-linear projection, $\mathbf{x}(t) = \wp(\boldsymbol{\zeta}(t))$ in 3.1, we use a linear projection $\mathbf{x}(t) = \mathbf{V}\boldsymbol{\zeta}(t)$. This is motivated by the fact that the linear projections are successfully utilized even in nonlinear model reduction. As a result of this, PDE (3.32) becomes:

$$\mathbf{f}\left(\mathbf{V}(\boldsymbol{\zeta}(t)), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t))\right) - \mathbf{E}\mathbf{V}\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) = \mathbf{0}. \tag{4.1}$$

   where $\boldsymbol{\varpi}(\boldsymbol{\zeta}(t))$ and $\boldsymbol{\varrho}(\boldsymbol{\zeta}(t))$ are user-defined, and $\mathbf{V} \in \mathbb{R}^{n\times r}$ is the desired solution.

2. **Column-wise consideration:** Using a linear projection $\mathbf{V}$ makes the system (4.1) under-determined since n equations are to be solved for $n \times r$ unknowns in $\mathbf{V} \in \mathbb{R}^{n\times r}$. This problem is avoided by considering the equation (4.1) column-wise for each $\mathbf{v}_i \in \mathbb{R}^n, i = 1, ..., r$, given as:

$$\mathbf{f}\left(\mathbf{v}_i(\boldsymbol{\zeta}_i(t), \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t))\right) - \mathbf{E}\mathbf{v}_i\boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t)) = \mathbf{0}, \tag{4.2}$$

   where $\boldsymbol{\zeta}_i(t) \in \mathbb{R}^r$ and $\mathbf{V} = [v_1, ..., v_r]$.

Kindly note that the column-wise consideration has the limitation that (4.1) is generally not fulfilled, since the *couplings* in $\mathbf{V}\boldsymbol{\zeta}(t), \mathbf{V}\boldsymbol{\varpi}(\boldsymbol{\zeta}(t))$ and $\boldsymbol{\varrho}(\boldsymbol{\zeta}(t))$ are not being considered. Hence, the column-wise consideration constitutes a further simplification/approximation step, which is naturally restricting the universality of the original nonlinear moment matching framework of [26].

3. **Time discretization:** Unlike the linear case, where the Sylvester equation (cf. 2.75) turns out to be a constant linear matrix equation, the nonlinear Sylvester equation (cf. 3.32) is a state-dependent equation, i.e., the state-vector $\boldsymbol{\zeta}(t)$ cannot be factored out. Hence, a time-discretized version of 3.32 is solved instead with time snapshots $[t_k], k = 1, .., K$, similar as in POD. Consequently, using a time-discrete version of the nonlinear signal generator $(\boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t_k)), \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t_k)), \boldsymbol{\zeta}_{0,i})$ 4.2 becomes:

$$\mathbf{f}\left(\mathbf{v}_{ik}(\boldsymbol{\zeta}_i(t_k), \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t_k)))\right) - \mathbf{E}\mathbf{v}_{ik}\boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t_k)) = \mathbf{0}. \tag{4.3}$$

Note that the discrete solution of $\boldsymbol{\zeta}_i(t_k)$ of the nonlinear signal generator ODE must be given or computed via simulation before solving (4.3).

Linear Signal Generator

One can also try to interconnect the non-linear system (3.1) with a linear signal generator (2.76), where $\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) = \boldsymbol{\Xi}\boldsymbol{\zeta}(t)$ and $\boldsymbol{\varrho}(\boldsymbol{\zeta}(t)) = \boldsymbol{\Psi}\boldsymbol{\zeta}(t)$ with $\boldsymbol{\Xi} = \mathbb{C}^{r\times r}, \boldsymbol{\Psi} \in \mathbb{C}^{m\times r}$.

1. **Linear projection:** Using a linear projection (4.1) becomes:

$$\mathbf{f}(\mathbf{V}(\boldsymbol{\zeta}(t), \boldsymbol{\Psi}(\boldsymbol{\zeta}(t)) - \mathbf{E}\mathbf{V}\boldsymbol{\Xi}\boldsymbol{\zeta}(t) = \mathbf{0}, \tag{4.4}$$

where the triple $(\boldsymbol{\Xi}(\boldsymbol{\zeta}(t)), \boldsymbol{\Psi}(\boldsymbol{\zeta}(t), \boldsymbol{\zeta}_0)$ is user defined. Note that connecting a linear signal generator (2.76) with a nonlinear system corresponds to exciting the nonlinear system with exponential inputs given as: $\mathbf{u}(t) = \boldsymbol{\Psi}e^{\boldsymbol{\Xi}t}\boldsymbol{\zeta}_\mathbf{0}$.

2. **Column-wise consideration:** A column-wise consideration of 4.4 yields:

$$\mathbf{f}\left(\mathbf{v}_i\boldsymbol{\zeta}_i(t), \boldsymbol{\Psi}_i(\boldsymbol{\zeta}_i(t))\right) - \mathbf{E}\mathbf{v}_i\boldsymbol{\Xi}_i\boldsymbol{\zeta}_i(t) = \mathbf{0}. \tag{4.5}$$

3. **Time discretization:** Using the time-discrete signal generator $\boldsymbol{\Xi}_i\boldsymbol{\zeta}_i(t_k)$, $\boldsymbol{\Psi}_i\boldsymbol{\zeta}_i(t_k)$ in (4.5) we have:

$$\mathbf{f}\left(\mathbf{v}_{ik}(\boldsymbol{\zeta}_i(t_k), \boldsymbol{\Psi}_i(\boldsymbol{\zeta}_i(t_k)))\right) - \mathbf{E}\mathbf{v}_{ik}\boldsymbol{\Xi}_i\boldsymbol{\zeta}_i(t_k) = \mathbf{0}. \tag{4.6}$$

Again note that the discrete solution of $\boldsymbol{\zeta}_i(t_k)$ of the linear signal generator ODE is analytically given by exponential functions with exponents $\boldsymbol{\Xi}_i$ to simulate the linear signal generator.

Zero Signal Generator

Finally, one can also use a zero signal generator defined as:

$$\dot{\boldsymbol{\zeta}}(t) = \boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) = \mathbf{0}, \tag{4.7a}$$
$$\mathbf{u}(t) = \boldsymbol{\varrho}(\boldsymbol{\zeta}_\mathbf{0}). \tag{4.7b}$$

This is a special case where $\boldsymbol{\zeta}(t) = \boldsymbol{\zeta}_\mathbf{0}$ and $\mathbf{u}(t) =$ constant. Connecting such a signal generator to a nonlinear system corresponding to exciting the nonlinear system with constant input functions.

1. **Linear projection:** For this case, (4.1) becomes

$$\mathbf{f}\left(\mathbf{V}\boldsymbol{\zeta}_\mathbf{0}, \boldsymbol{\varrho}(\boldsymbol{\zeta}_\mathbf{0})\right) = \mathbf{0}. \tag{4.8}$$

2. **Column-wise consideration:** This operation yields

$$\mathbf{f}\left(\mathbf{v}_i\boldsymbol{\zeta}_{\mathbf{0}i}, \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_{\mathbf{0}i})\right) = \mathbf{0}, \tag{4.9}$$

   with $\dot{\boldsymbol{\zeta}}_i(t) = \mathbf{0}, \mathbf{u}_i(t) = \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_{\mathbf{0}i}) =$ constant, and $\boldsymbol{\zeta}_i(t) = \boldsymbol{\zeta}_{\mathbf{0}i} =$ constant for $i = 1, .., r$.

3. **Time discretization:** Since (4.9) represents a state-independent equation, hence no time-discretization is needed for this case.

Now using these simplification for the nonlinear Sylvester PDE (3.32), the algorithm to obtain the orthonormal basis via nonlinear moment-matching is presented below [296]:

4.1.2 Algorithm for Nonlinear Moment Matching

Algorithm 4 Nonlinear Moment Matching (NLMM): For offline basis generation

INPUT: $\mathbf{f}(\mathbf{x}, \mathbf{u}), \mathbf{J}(\mathbf{x}, \mathbf{u}), \boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t_k)), \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t_k)), \boldsymbol{\zeta}_i(t_k)$, initial guess $\mathbf{v}_{0,ik}$, reduced rank order r

OUTPUT: Orthogonal basis: $\mathbf{V}_{\text{NLMM}}$

for $i = 1 : r$ **do**
  for $k = 1 : K$ **do**
    $fun = @(\mathbf{v})\ \mathbf{f}(\mathbf{v} * \boldsymbol{\zeta}_{i,k}, \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_{i,k})) - \mathbf{E}\mathbf{v} * \boldsymbol{\varpi}_i(\boldsymbol{\zeta}_{i,k})$ ▷ residual
    $Jfun = @(\mathbf{v})\ \mathbf{J}(\mathbf{v} * \boldsymbol{\zeta}_{i,k}, \boldsymbol{\varrho_i}(\boldsymbol{\zeta}_{i,k}))\boldsymbol{\zeta}_{i,k} - \mathbf{E}\boldsymbol{\varpi}_i(\boldsymbol{\zeta}_{i,k})$ ▷ Jacobian of residual
    $\mathbf{v}_{ik} = \mathbf{NewtonRaphson}(fun, \mathbf{v}_{0,ik}, Jfun)$
    $\mathbf{V}_{\text{NLMM}}(:, (i-1) * K + k) \leftarrow \mathbf{v}_{ik}$
    $\mathbf{V}_{\text{NLMM}} = \mathbf{GramSchmidt}(\mathbf{v}_{ik}, \mathbf{V}_{\text{NLMM}})$ ▷ optional
  end for
end for
$[\mathbf{U},\ \boldsymbol{\Sigma}, \tilde{}] = \mathbf{SVD}(\mathbf{V}_{\text{NLMM}})$; $\mathbf{V}_{\text{NLMM}} = \mathbf{U}(:, 1 : r)$

The above algorithm is used to generate the basis $\mathbf{V}_{\text{NLMM}}$ for the reduced subspace. Note that we have presented the most general form of the algorithm, whereby a nonlinear signal generator has been employed. One can use a suitable linear or zero signal generator as well. Consequently, lines 3 and 4 of Algorithm 4 will change. The algorithm employs two nested **for** loops to simulate each signal generator $i = 1, .., r$ at each collocation time shift $k = 1, .., K$. $Jfun$ denotes the analytical Jacobian of the nonlinear function fun, which is provided to speed up the process.

NLMM is "simulation-free" in the sense that it does not require the numerical integration of the large-scale nonlinear system (3.1). However, it involves the solution of (at most) $r.K$ nonlinear systems of equations (NLSE) of full order dimension n. These are solved using an implicit *Newton Raphson* scheme (cf. line 5, Algorithm 4) to generate the columns of $\mathbf{V}_{\text{NLMM}}$. Finally, the redundant columns of $\mathbf{V}_{\text{NLMM}}$ are truncated via the SVD scheme. This is in contrast with the POD method whereby to simulate the FOM, an NLSE has to be solved implicitly at each time step which results in increased computational costs.

4.2 NLMM-DEIM

Now, we are ready to introduce the first practical reduction framework for nonlinear systems, i.e., the nonlinear moment-matching with discrete empirical interpolation method (NLMM-DEIM). While NLMM avoids the expensive simulation of the underlying nonlinear Sylvester PDE (3.32), as illustrated in the previous section, the DEIM (Algorithm 3) further reduces the online cost of evaluation of nonlinear terms. This substantially reduces the overall computational costs to obtain the orthonormal projection basis. The numerical algorithm for this framework is as follows:

Algorithm 5 NLMM-DEIM

Input: FOM $\dot{\boldsymbol{x}}(t)$, Signal Generator $\dot{\boldsymbol{\zeta}}(t)$, r : approximating rank for NLMM, and m_d number of DEIM modes.

1: Compute $\mathbf{V}_{\text{NLMM}}$ following Algorithm 4 ▷ orthogonal basis
2: Obtain function snapshots $\mathbf{F} = [\mathbf{f}(\mathbf{x}(t_1))\ \mathbf{f}(\mathbf{x}(t_2))... \ \mathbf{f}(\mathbf{x}(t_{n_s}))]$
3: Calculate $[\mathbf{u}_l,\ \mathbf{\Sigma_d}, \tilde{}] = \mathbf{SVD}(\mathbf{F})$ ▷ nonlinear snapshots
4: Obtain $\mathbf{U_d}, \mathbf{P}$ and $\vec{p}$ from Algorithm 3 ▷ call DEIM
5: Construct the reduced model (3.10) with the reduced nonlinear function obtained from (3.47)
6: Integrate (3.10)
7: Project back full solution

4.2.1 Case studies

We will now demonstrate the performance of the proposed NLMM-DEIM framework. For that, we consider the two nonlinear benchmark models: the nonlinear viscous Burgers' equation and the FitzHugh-Nagumo model. Before we proceed with the numerical validation, we enlist some details concerning all the numerical models presented in this thesis:

1. All the numerical simulations are performed in-silico on MATLAB® R2019a version 9.6.0 running on Intel Core™ i7-4790 CPU @3.60GHz with 12GB RAM.

2. All ROMs (except the interpolated ones) are obtained via the one-sided Galerkin projection with $\mathbf{W} = \mathbf{V}$ and the local basis are made orthogonal, i.e. $\mathbf{V^T V} = \mathbf{I}$.

3. All the FOMs and ROMs are integrated using the implicit Euler's scheme within the Newton Raphson method having a relative tolerance of 10^{-3}.

Nonlinear viscous Burgers' equation

Burgers' equation is studied under various areas of applied mathematics such as gas dynamics, fluid mechanics, traffic flow, non-linear acoustics, theory of shock waves, and in continuous stochastic processes. Due to the presence of viscosity and convection terms, it's structure roughly matches with *Navier-Stokes* equation.

The one-dimensional, parabolic, quasi-linear, viscous Burgers' equation is given as:

$$\frac{\partial \mathbf{u}}{\partial t}(\mathbf{x},t) + \mathbf{u}(\mathbf{x},t)\frac{\partial \mathbf{u}}{\partial \mathbf{x}}(\mathbf{x},t) = \frac{1}{Re}\left(\frac{\partial^2 \mathbf{u}}{\partial \mathbf{x}^2}(\mathbf{x},t)\right) \quad \mathbf{x} \in (0,L), t \in [0,T], \tag{4.10}$$

subject to the following initial and boundary conditions:

$$\mathbf{u}(\mathbf{x},0) = 0.1, \tag{4.11a}$$

$$\mathbf{u}(0,t) = \mathbf{r}(t);\ \frac{d\mathbf{u}}{d\mathbf{x}}(L,t) = 0, \tag{4.11b}$$

where Re represents the Reynold's number and influences the effect of diffusion. For our evaluations, we selected Re = 100 and the length of spatial domain $L = 1$. After a semi-discretization using finite-difference scheme with n grid points, we obtained the following quadratic-bilinear structure of the state-space model:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{H}\ (\mathbf{x}(t) \otimes \mathbf{x}(t)) + (\mathbf{N}\mathbf{x}(t) + \mathbf{B})\mathbf{r}(t), \tag{4.12}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ and $(\mathbf{x} \otimes \mathbf{x})$ is the *Kronecker* product. The matrices $\mathbf{A}, \mathbf{H}$, and $\mathbf{N}$ can be seen in Appendix (A.1). A single control input was applied at a time on the left boundary, and the FOM was solved using the implicit Euler's method.

To proceed with the ROM via NLMM-DEIM, Algorithm 5 was employed. Another ROM was obtained via POD for the sake of comparison. Note that NLMM was trained offline for the inputs generated by the chosen signal generator. However, in the online evaluation of the ROM, two different test inputs were used $\mathbf{r}_1(t)$, and $\mathbf{r}_2(t)$. In the case of POD, the training and the test inputs were kept the same. This deliberate choice of inputs was made to test the true potential of NLMM-DEIM. The various parameters used for this test case are presented in Table 4.1.

The output response and the error plot of the full and the ROMs obtained via NLMM-DEIM are presented in Fig. 4.1 for the two test signals. As can be seen, the ROMs faithfully capture the dynamics of the original system for both the test cases. Quantitatively, the comparison of computational times for the offline and online phase are presented in Table 4.2 along with the approximating errors in Table 4.2. The comparison shows that NLMM requires considerably less time than POD to compute the projection basis for both the test cases and for different choices of reduced dimensions. As far as the online evaluations of the ROMs are considered, the DEIM has further reduced the simulation times by efficiently evaluating the underlying nonlinear terms for both cases.

Table 4.1: Burgers' equation: Parameters of the selected scenarios

Method	Parameters
Full model	$n = 500, L = 1, t \in [0, 30s], Re = 100, h = 0.01s$
NLMM	$r = (10, 20), \dot{\boldsymbol{\zeta}}(t) = \tanh(t) + 5e^{-4}, i = 1, K = 51$
DEIM	$m_d = (10, 20), n_s = 250$
POD	$r = (10, 20)$
Test Inputs	$\mathbf{r}_1(t) = 0.5(\cos(2\pi/10)t), \mathbf{r}_2(t) = \text{sign}(t)$

Table 4.2: Burgers' equation: Computational times for both test cases

method	size	offline time (s) (test 1)	online time (s) (test 1)	offline time (s) (test 2)	online time (s) (test 2)
FOM	500	-	559.69	-	208.57
POD	10	543.89 (300 NLSE)	7.931	209.81	5.323
POD	20	543.92 (300 NLSE)	13.65	209.81	8.549
NLMM-DEIM	10	8.346 (51 NLSE)	2.452	3.345	1.312
NLMM-DEIM	20	11.331 (51 NLSE)	4.633	5.358	1.531

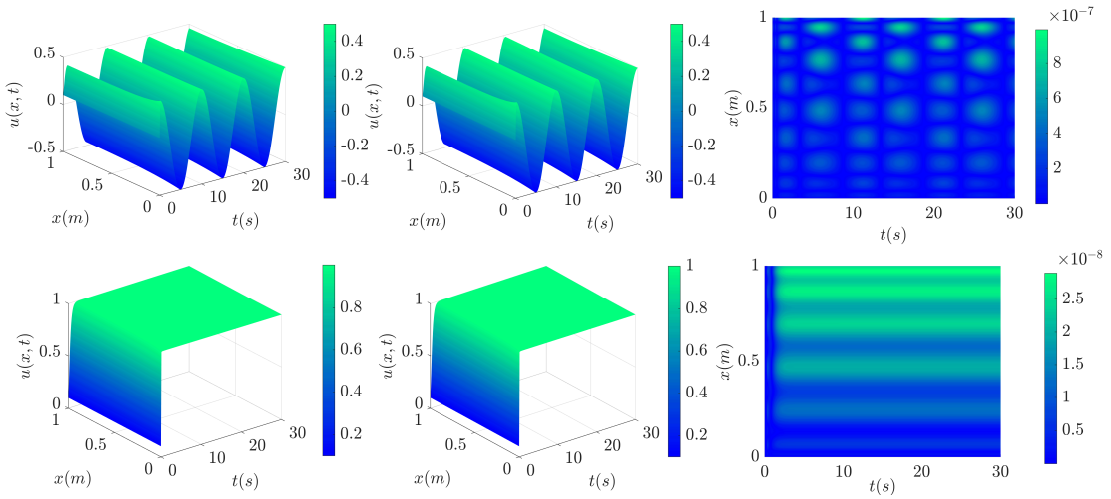

Figure 4.1: Burgers' equation: Top row: input $\mathbf{r}(t) = 0.5(\cos(2\pi/10)t)$, bottom row: input $\mathbf{r}(t) = \text{sign}(t)$, (left) full order model, (middle) NLMM-DEIM ROM, (right) point-wise relative error of ROM

Table 4.3: Burgers' equation: Approximation errors for both test cases

input	norm	POD		NLMM-DEIM	
		$r = 10$	$r = 20$	$r = 10$	$r = 20$
$\mathbf{r}_1(t)$	$\mathcal{L}_1$	$1.04e^{-9}$	$1.87e^{-11}$	$9.20e^{-5}$	$8.60e^{-7}$
	$\mathcal{L}_2$	$8.13e^{-11}$	$1.37e^{-12}$	$7.24e^{-6}$	$6.91e^{-8}$
	$\mathcal{L}_\infty$	$2.47e^{-11}$	$1.73e^{-13}$	$9.9e^{-7}$	$8.74e^{-9}$
$\mathbf{r}_2(t)$	$\mathcal{L}_1$	$7.64e^{-9}$	$8.8e^{-4}$	$1.58e^{-9}$	$5.75e^{-6}$
	$\mathcal{L}_2$	$5.75e^{-10}$	$5.68e^{-5}$	$1.21e^{-10}$	$3.70e^{-7}$
	$\mathcal{L}_\infty$	$4.34e^{-11}$	$3.69e^{-6}$	$9.31e^{-12}$	$2.41e^{-8}$

Fitz-Hugh Nagumo model

The FitzHugh-Nagumo (FHN) model is a simplified version of the Hodgkin-Huxley model. The FHN model is used to describe the activation and deactivation dynamics of a spiking neuron. This model has been extensively used in literature for POD [80] and quadratic-bilinear MOR [62, 131]. The underlying dynamics are governed by a coupled PDE-ODE system given as:

$$\epsilon \frac{\partial \mathbf{v}}{\partial t}(\mathbf{x}, t) = \epsilon^2 \frac{\partial^2 \mathbf{v}}{\partial \mathbf{x}^2}(\mathbf{x}, t) + \mathbf{f}(\mathbf{v}(\mathbf{x}, t)) - \boldsymbol{\omega}(\mathbf{x}, t) + g, \tag{4.13a}$$

$$\frac{\partial \boldsymbol{\omega}}{dt}(\mathbf{x}, t) = R\mathbf{v}(\mathbf{x}, t) - \gamma \boldsymbol{\omega}(\mathbf{x}, t) + g, \tag{4.13b}$$

where the nonlinear function $\mathbf{f}(\mathbf{v}) = \mathbf{v}(\mathbf{v} - 0.1)(1 - \mathbf{v})$ with the initial and boundary condition defined as:

$$\mathbf{v}(\mathbf{x}, 0) = 0, \qquad \boldsymbol{\omega}(\mathbf{x}, 0) = 0, \qquad \mathbf{x} \in [0, L], \tag{4.14a}$$

$$\frac{\partial \mathbf{v}}{dt}(0, t) = -i_0(t), \qquad \frac{d\mathbf{v}}{d\mathbf{x}}(L, t) = 0, \qquad t \in [0, T]. \tag{4.14b}$$

A spatial discretization of the underlying coupled nonlinear PDE into l elements yields an input-affine state-space model of $n = 2l$ degrees of freedom. The model equation is given by:

$$\begin{aligned} \mathbf{E}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{f}(\mathbf{x}(t)) + \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t), \end{aligned} \tag{4.15}$$

where the state variables $\mathbf{x} = [\mathbf{v}^T \ \boldsymbol{w}^T]$ represent the voltage and recovery voltage at each spatial element. The control input $\mathbf{u}(t) = [i_0(t), 1]^T$ is applied on the left boundary ($\mathbf{x} = 0$), and the desired output was measured on the right boundary ($\mathbf{x} = L$). Depending upon the amplitude of the input source $i_0(t)$, different neuronal behaviors can occur. For example when the stimulus $i_0(t)$ exceeds a certain threshold, a resting behavior (limit-cycle oscillations) can be observed. Similarly, a saturated behavior is recorded for higher currents. Depending upon the nature of the system, the various parameters of the FOM, POD and NLMM selected are enlisted in Table 4.4.

For the application of NLMM, a single signal generator was selected by pre-analyzing the value range that the signal $\boldsymbol{\zeta}(t)$ covers in the time span $t_k \in [0, 5s]$. For DEIM, 500 snapshots of the nonlinear terms were captured, and the reduced basis for the nonlinear function $\mathbf{f}()$ was constructed. The numerical results are presented in Figs. 4.2 and 4.3 whereas the approximation errors and simulation times are enlisted in Tables 4.5 and 4.6 respectively.

As expected, NLMM required lesser computational times to obtain the reduction basis than POD. This is because of the lower number of NLSEs that need to be solved in NLMM. On the other hand, POD produced more accurate solutions than NLMM due to extensive information captured via snapshots during the training phase. However, POD needed several different training simulations to capture other neuron behaviors like resting, spiking, and blocking, which NLMM selectively captured without the expensive offline simulations. Regarding the online phase, the speed-up gained through dimension reduction was further enhanced via DEIM.

Table 4.4: FHN model: Parameters of the selected scenarios

Method	Parameters
Full model	$n = 2000, L = 1, t \in [0, 15s], \epsilon = 0.015, R = 0.5, \gamma = 2, g = 0.05$
NLMM	$r = (22, 33), \dot{\boldsymbol{\zeta}}(t) = \boldsymbol{\zeta}(t) + 0.3, i = 1, K = 41$
DEIM	$m_d = (25), n_s = 500$
POD	$r = (22, 33)$
Test Inputs	$i_0(t) = 5.10^4 t^3 e^{-15t}$

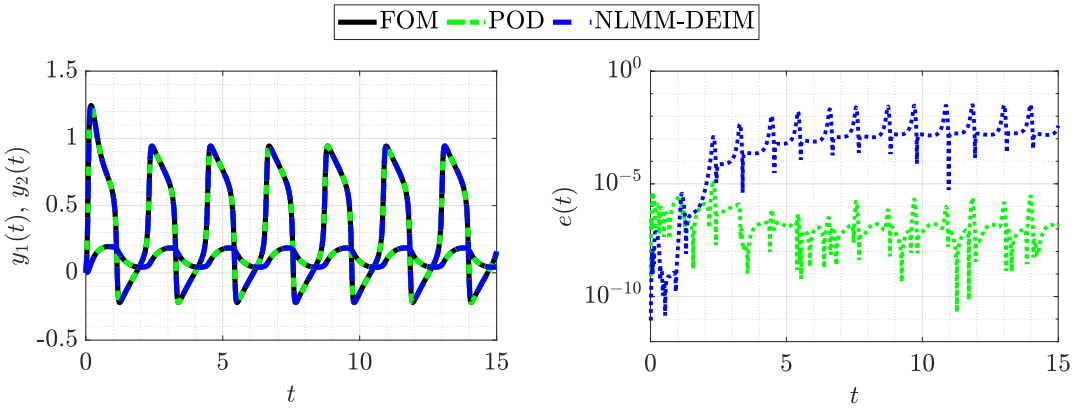

Figure 4.2: FHN model: Comparison of the output responses (left) output responses (right) point-wise relative error

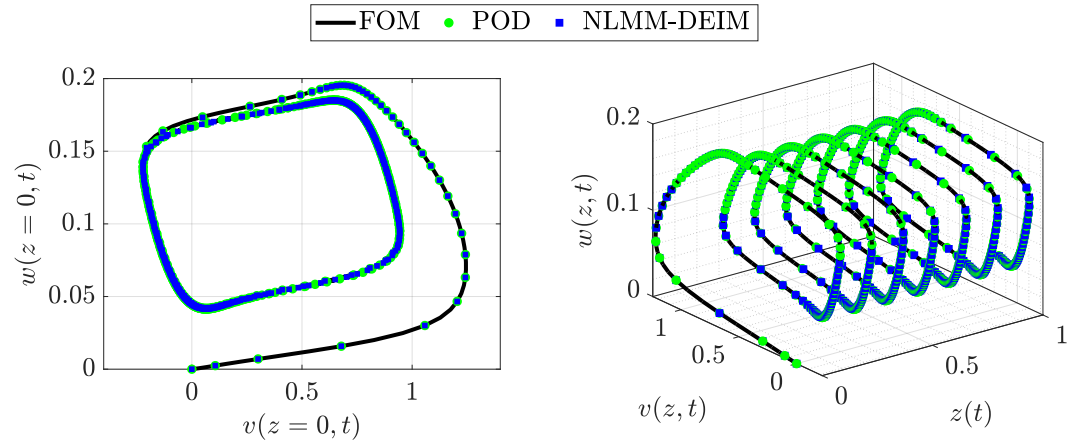

Figure 4.3: FHN model: spiking: (left) phase portrait, (right) limit cycle behavior

Table 4.5: FHN model: approximation errors

output	norm	POD		NLMM-DEIM	
		$r = 22$	$r = 33$	$r = 22$	$r = 33$
$\mathbf{y}_1(t)$	$\mathcal{L}_1$	$5.8e^{-4}$	$6.28e^{-9}$	$3.20e^{-6}$	$2.13e^{-7}$
	$\mathcal{L}_2$	$4.26e^{-5}$	$4.66e^{-10}$	$2.24e^{-5}$	$1.30e^{-8}$
	$\mathcal{L}_\infty$	$1.45e^{-5}$	$2.15e^{-10}$	$3.3e^{-7}$	$2.28e^{-9}$
$\mathbf{y}_2(t)$	$\mathcal{L}_1$	$1.16e^{-3}$	$1.78e^{-5}$	$7.3e^{-3}$	$3.36e^{-5}$
	$\mathcal{L}_2$	$4.32e^{-4}$	$7.39e^{-7}$	$2.3e^{-3}$	$1.35e^{-5}$
	$\mathcal{L}_\infty$	$6.46e^{-5}$	$1.06e^{-7}$	$1.4e^{-3}$	$1.41e^{-7}$

Table 4.6: FHN model: computational times

method	size	offline time (s)	online time (s)
FOM	2000	-	551.21
POD	22	543.45 (500 NLSE)	51.45
POD	33	543.45 (500 NLSE)	57.19
NLMM-DEIM	22	21.20 (41 NLSE)	5.65
NLMM-DEIM	33	24.01 (41 NLSE)	7.30

4.3 NLMM-DMD

The NLMM-DEIM framework described in previous section works satisfactorily for a variety of nonlinear systems [237, 242, 238, 239]. However, the scheme can be considered as a *semi non-intrusive* MOR approach. This means that it requires the solution of the nonlinear function $\mathbf{f}(\mathbf{x}, \mathbf{u})$ and the Jacobian function $\mathbf{A}(\mathbf{x}, \mathbf{u})$ within the Newton Raphson scheme to obtain the orthonormal basis vectors without the need for an analytical expression of the governing equations (3.1). These can rather can be evaluated within any commercial FE software package and hence obtain $\mathbf{V}_{\text{NLMM}}$. However, for the evaluation of the ROM via projection, the governing equations are needed, making the scheme semi non-intrusive. Presently, the research community has increased interest in developing non-intrusive-based reduction methods that aim to learn the reduced models from data. This enables the application of MOR techniques where the operators of high-dimensional systems are unavailable. In what follows, we describe how dynamic mode decomposition can be used as a complete non-intrusive method to reduce the underlying nonlinear functions.

4.3.1 Dynamic Mode Decomposition

Dynamic mode decomposition (DMD) stems from the works of *Bernard Koopman* [170] in 1931, which was later revived in the studies presented by Mezic et al. [207, 208, 209]. This method was originally used to decompose complex flows into simple spatial-temporal coherent structures. Schmid [269, 270] first defined the DMD algorithm, and demonstrated its applications to high-dimensional fluid data. This method is used to find the best-fit linear operator (aka *Koopman operator*) that progresses high-dimension measurements forward in time [288]. More specifically, DMD estimates the modes of *Koopman* operator which is an infinite-dimensional linear operator that represents the flow of a nonlinear dynamical system on the *Hilbert* space of measurement functions of states. DMD can be defined as:

Definition 4.1 (Dynamic Mode Decomposition) *[176]*

Given a dynamical system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)), \tag{4.16}$$

with two sets of data:

$$\mathbf{X_1} = \begin{bmatrix} | & | & & | \\ \mathbf{x}(t_0) & \mathbf{x}(t_1) & \cdots & \mathbf{x}(t_{k-1}) \\ | & | & & | \end{bmatrix},$$

$$\text{and } \mathbf{X_2} = \begin{bmatrix} | & | & & | \\ \mathbf{x}(t_1) & \mathbf{x}(t_2) & \cdots & \mathbf{x}(t_k) \\ | & | & & | \end{bmatrix},$$

where $\mathbf{f}$ is a nonlinear flow map, and $\mathbf{x}(t_1), \mathbf{x}(t_2)..., \mathbf{x}(t_k)$ represent the snapshots measurements at $t = \{1, 2, ..., k\}$. DMD computes the leading eigendecomposition of the best-fit linear operator $\mathbf{A}$ relating the data $\mathbf{X_2} \approx \mathbf{A}\mathbf{X_1}$, with:

$$\mathbf{A} = \mathbf{X_2}\mathbf{X_1}^{\dagger}, \tag{4.17}$$

and where † denotes the *Moore-Penrose* pseudoinverse. ▲

This best-fit linear operator characterizes a linear dynamical system that advances snapshot measurements forward in time i.e;

$$\mathbf{x}(k+1) \approx \mathbf{A}\mathbf{x}_k, \tag{4.18}$$

Also, $\mathbf{A}$ can mathematically be defined as:

$$\mathbf{A} = \underset{\mathbf{A}}{\operatorname{argmin}} \|\mathbf{X_2} - \mathbf{A}\mathbf{X_1}\|_F = \mathbf{X_2}\mathbf{X_1}^{\dagger} \tag{4.19}$$

Usually, the operator $\mathbf{A}$ is ill-conditioned for large-scale systems and in practice, DMD computes the eigendecomposition of $\mathbf{A}$ by a reduced rank representation $\tilde{\mathbf{A}}$. As a result, the following proxy for (4.16) is established:

$$\dot{\tilde{\mathbf{x}}}(t) = \tilde{\mathbf{A}}\tilde{\mathbf{x}}(t), \tag{4.20}$$

whose solution is given as

$$\tilde{\mathbf{x}}(t) = \sum_{i=1}^{m} \mathbf{b}_i \boldsymbol{\Psi}_i e^{(\boldsymbol{\omega}_i t)}, \tag{4.21}$$

where m_D is the reduced DMD rank, $\boldsymbol{\Psi}_i, \boldsymbol{\omega}_i$ are the eigenfunctions of $\tilde{\mathbf{A}}$ and $\mathbf{b}_i$ are the initial conditions. The procedure is described as follows [288]:

1. Obtain the SVD of $\mathbf{X_1}$

   $$\mathbf{X_1} \approx \hat{\mathbf{U}}\hat{\mathbf{\Sigma}}\hat{\mathbf{V}}^*,$$

   with $\hat{\mathbf{U}} \in \mathbb{C}^{n \times m_D}, \hat{\mathbf{\Sigma}} \in \mathbb{C}^{m_D \times m_D}, \hat{\mathbf{V}} \in \mathbb{C}^{k \times m_D}$ $(m_D \leq k)$ and $*$ represents the conjugate transpose operation.

2. Now compute the full matrix $\mathbf{A}$ according to (4.17)

   $$\mathbf{A} = \mathbf{X_2}\hat{\mathbf{V}}\hat{\mathbf{\Sigma}}^{-1}\hat{\mathbf{U}}^*,$$

   and consequently, $\tilde{\mathbf{A}}$ can be obtained by projecting $\mathbf{A}$ onto the left singular vectors $\hat{\mathbf{U}}$

   $$\tilde{\mathbf{A}} = \hat{\mathbf{U}}^*\mathbf{A}\hat{\mathbf{U}} = \hat{\mathbf{U}}^*\mathbf{X_2}\hat{\mathbf{V}}\hat{\mathbf{\Sigma}}^{-1}.$$

3. Next, evaluate the spectral decomposition of $\tilde{\mathbf{A}}$ as

   $$\tilde{\mathbf{A}}\boldsymbol{\Upsilon} = \boldsymbol{\Upsilon}\boldsymbol{\Lambda},$$

   where the DMD eigenvalues are the diagonal entries of $\boldsymbol{\Lambda}$ which also correspond to eigenvalues of $\mathbf{A}$, and the eigenvectors of $\tilde{\mathbf{A}}$ are the columns of $\boldsymbol{\Upsilon}$.

4. Finally, the high-dimensional DMD modes $\mathbf{V}_{\text{DMD}}$ are obtained using the eigenvectors $\boldsymbol{\Upsilon}$ and time-displaced snapshot matrix $\mathbf{X_2}$, given as:

   $$\mathbf{V}_{\text{DMD}} = \mathbf{X}_2\hat{\mathbf{V}}\hat{\mathbf{\Sigma}}^{-1}\boldsymbol{\Upsilon}.$$

   The pseudo-code summarizing the DMD procedure is given in Algorithm (6).

Algorithm 6 DMD Mode Extraction

Input: Snapshots $\{\mathbf{x}(t_0), \mathbf{x}(t_1), .., \mathbf{x}(t_k)\}$
Output: Projection Matrix $\mathbf{V}_{\text{DMD}}$

1: Set $\mathbf{X}_1 = [\mathbf{x}(t_0), ..., \mathbf{x}(t_{k-1})]$, $\mathbf{X}_2 = [\mathbf{x}(t_1), ..., \mathbf{x}(t_k)]$
2: $\hat{\mathbf{U}}, \hat{\mathbf{\Sigma}}, \hat{\mathbf{V}}^* = \mathbf{SVD}(\mathbf{X}_1)$
3: Define $\tilde{\mathbf{A}} = \hat{\mathbf{U}}^*\mathbf{X}_2\hat{\mathbf{V}}\hat{\mathbf{\Sigma}}^{-1}$
4: Solve the eigenvalue problem: $[\boldsymbol{\Upsilon}, \boldsymbol{\Lambda}] = eig(\tilde{\mathbf{A}})$
5: $\mathbf{V}_{\text{DMD}} = \mathbf{X}_2\hat{\mathbf{V}}\hat{\mathbf{\Sigma}}^{-1}\boldsymbol{\Upsilon}$

DMD can also be used as a *Galerkin* projection framework (see [5]). However, this may additionally require the orthogonalization of the DMD modes using a *Gram-Schmidt* procedure.

4.3.2 Application of DMD in dimensionality reduction

In the following, we will demonstrate how DMD can can be used to obtain low-rank approximations of the nonlinear function as a substitute over DEIM. Consider the nonlinear snapshots $\{\mathbf{f}(\mathbf{x}(t_0)), \mathbf{f}(\mathbf{x}(t_1)), .., \mathbf{f}(\mathbf{x}(t_k))\}$ taken from (3.1). Then, the DMD approximation of the nonlinear function $\mathbf{f}(\mathbf{x}(t))$ via Algorithm (6) reads:

$$\mathbf{f}_{\text{DMD}}(\mathbf{x}(t)) = \sum_{i=1}^{m} \mathbf{b}_i \mathbf{\Psi}_i e^{(\boldsymbol{\omega}_i t)} \approx \mathbf{V}_{\text{DMD}} \text{diag}(e^{\boldsymbol{\omega}_{\text{DMD}} t}) \mathbf{b}, \tag{4.22}$$

where $\mathbf{b} = (\mathbf{V}_{\text{DMD}})^{\dagger} \mathbf{f}(\mathbf{x}(t_0)) \in \mathbb{R}^{m_D}$ Now using this approximation in (3.10) we obtain the NLMM-DMD reduced model given as:

$$\begin{aligned} \dot{\mathbf{x}}_r(t) &= {\mathbf{V}_{\text{NLMM}}}^T \mathbf{V}_{\text{DMD}} \text{diag}(e^{\boldsymbol{\omega}_{\text{DMD}} t}) \mathbf{b} + {\mathbf{V}_{\text{NLMM}}}^T \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}_r(t) &= \mathbf{g}(\mathbf{V}_{\text{NLMM}} \mathbf{x}_r(t)), \end{aligned} \tag{4.23}$$

where $(\mathbf{V}_{\text{NLMM}})^T \mathbf{V}_{\text{DMD}} \in \mathbb{R}^{r \times m_D}$ and $\text{diag}(e^{\boldsymbol{\omega}_{\text{DMD}} t}) \in \mathbb{R}^{m_D}$.

As we can see that the size of this new ROM (4.23) is independent of the full model as in the case of DEIM. Note that the nonlinear snapshots required for initializing DMD are needed only once. After the DMD approximation (4.22) is obtained, no further evaluation of the nonlinear term is required in the online stage yielding a significant speedup in the evaluation of the reduced model. This is also verified by the flop count requirements of the NLMM-DMD procedure as compared to POD-DEIM as presented in Table (4.7).

Table 4.7: Comparison of the computational complexity between POD-DEIM and NLMM-DMD

Procedure	Complexity
FULL	$\mathcal{O}(2n^2 + n)$
POD-DEIM	$\mathcal{O}(2r^2 + 4m_d r + r)$
NLMM-DMD	$\mathcal{O}(2r^2 + r)$

Furthermore note that the efficiency of DEIM is related to nonlinear term $\mathbf{f}$ and the DEIM approximation (cf 3.46) may not always hold, especially when the nonlinear components are not linearly independent. As we will show next, NLMM coupled with DMD gives an added computational savings than POD-DEIM while being a complete non-intrusive MOR approach. The overall scheme can be summed up in Algorithm (7).

Algorithm 7 NLMM-DMD

Input: FOM $\dot{\mathbf{x}}(t)$, Signal Generator $\dot{\boldsymbol{\zeta}}(t)$, r number of NLMM modes and m_D number of DMD modes.

1: Compute $\mathbf{V}_{\text{NLMM}}$ following Algorithm (4) ▷ orthogonal basis (NLMM)
2: Obtain function snapshots $\{\mathbf{f}(\mathbf{x}(t_0)), ..., \mathbf{f}(\mathbf{x}(t_k))\}$
3: Set $\mathbf{X}_1 = [\mathbf{f}(\mathbf{x}(t_0)), \mathbf{f}(\mathbf{x}(t_1)), .., \mathbf{f}(\mathbf{x}(t_{k-1}))]$
4: Set $\mathbf{X}_2 = [\mathbf{f}(\mathbf{x}(t_1)), ..., \mathbf{f}(\mathbf{x}(t_k))]$
5: Compute DMD modes from Algorithm (6) ▷ call DMD
6: Integrate (4.23)
7: Project back full solution

4.3.3 Case studies

To present the numerical validation of NLMM-DMD framework, we use two benchmark problems: Chafee-Infante equation and the nonlinear circuit example.

Chaffee-Infante model

First, we consider the one-dimensional Chafee-Infante also known as Allen-Cahn equation. This model has a cubic nonlinearity and models the reaction-diffusion in a variety of applications like chemical processes [87], fluid dynamics [74], and biological sciences [255]. The governing PDE is given as:

$$\frac{\partial \vartheta}{\partial t}(\mathbf{x}, t) = \frac{\partial^2 \vartheta}{\partial \mathbf{x}^2}(\mathbf{x}, t) - \lambda(\vartheta^3(\mathbf{x}, t) - \vartheta(\mathbf{x}, t)), \tag{4.24}$$

with initial condition and boundary conditions as:

$$\begin{aligned} &\vartheta(\mathbf{x}, 0) = 1 - \sin(\pi \mathbf{x}/L), && \mathbf{x} \in (0, L), \\ &\vartheta(0, t) = \mathbf{u}(t), \qquad \frac{\partial \vartheta}{\partial \mathbf{x}}(L, t) = 0, && t \in (0, T), \end{aligned}$$

where λ being the diffusion coefficient, adjusts the relative balance between the diffusion term and the nonlinear term. For our evaluations, we used $\lambda = 1$ and length of the spatial domain $L = 1$. A finite difference method with l grid points yielded a system of form $\mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{A}\mathbf{x} + \mathbf{f}(\mathbf{x} + \mathbf{b}\mathbf{u})$, $(\mathbf{E} = \mathbf{I})$ of dimension $n = l$. A single control input $\mathbf{u}(t)$ was applied on the left boundary $\mathbf{x} = 0$, and the output of interest was measured on the right boundary $\mathbf{x} = L$. The various parameters of FOM, NLMM, DMD, POD, and DEIM used for this model are enlisted in Table 4.8.

Table 4.8: Chafee-Infante model: Parameters used

Method	Parameters
Full model	$n = 1000, L = 1, \lambda = 1, t \in [0, 2s]$
NLMM	$r = 10, \dot{\boldsymbol{\zeta}}(t) = 0.5\sin(\pi t), i = 1, K = 61, t_k \in [0, 10s]$
DMD	$m_D = 10, k = 200$
POD	$r = 10, n_s = 200$
DEIM	$m_d = 10, n_s = 200$
Test Input	$\mathbf{u}(t) = 0.5(\cos(2\pi t) + 1)$

First, we constructed the POD-DEIM approximation for the sake of comparison. For that, the POD basis was obtained using 200 snapshots of the FD model of the Chafee-Infante PDE (4.24). Then, hyper-reduction using DEIM was carried with $m_d = 10$ interpolation indices via Algorithm 3. To obtain the projection basis via NLMM, a single signal generator $\dot{\boldsymbol{\zeta}}(t)$ was solved and used in the offline stage using Algorithm 4. For online evaluation, the DMD method was used (Algorithm 6) with 200 snapshots of the nonlinear term, that were already available from the DEIM. Note that for the NLMM-DMD framework, different inputs were used for training and testing (as explained earlier), whereas, for POD-DEIM, the training and test inputs were kept the same.

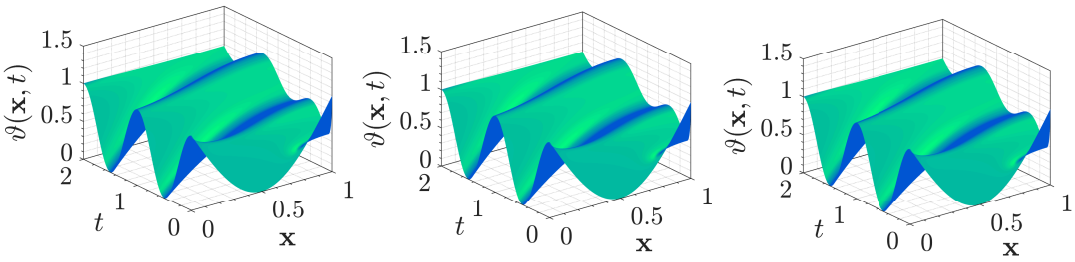

Figure 4.4: Chafee-Infante: Solution (left) full order (middle) POD-DEIM ROM (right) NLMM-DMD ROM

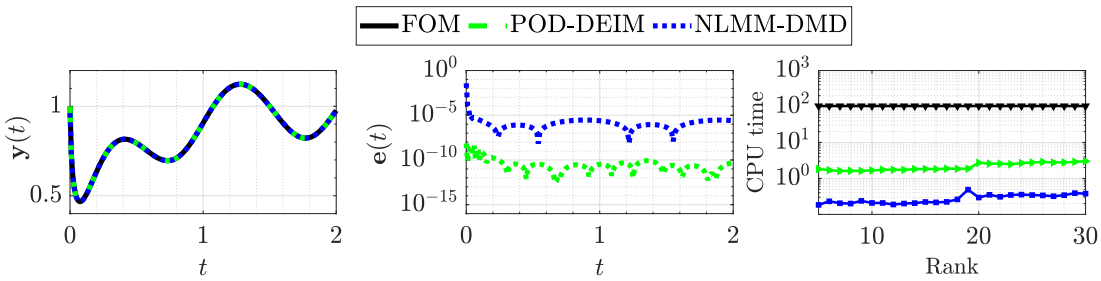

Figure 4.5: Chafee-Infante: Comparison of (left) output response, (middle) pointwise realtive error, (right) CPU times

Table 4.9: Chafee-Infante: Numerical results

Method	offline time (s)	online time (s)	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_\infty$
FOM	-	103.04	-		
POD-DEIM	103.09 (200 NLSE)	1.66	$1.67e^{-9}$	$3.7e^{-7}$	$6.3e^{-9}$
NLMM-DMD	44.70 (61 NLSE)	0.23	$2.23e^{-4}$	$3.1e^{-6}$	$3.3e^{-8}$

The numerical results are presented in Figs. 4.4 and 4.5. Moreover, the approximation errors and the CPU times are enlisted in Table 4.9. From the result, we can observe that POD-DEIM yields good approximation than NLMM-DMD. However, the latter yields a significant speed-up both in offline as well as online simulation times. Moreover, as seen from Fig. 4.5, the simulation time of DMD is always lesser as compared to DEIM for varying approximation ranks of the nonlinear function. This trend is expected since DMD avoids evaluating the nonlinear terms in the online stage, wheres DEIM requires the solution of reduced nonlinear inner products.

Nonlinear transmission-line model

Finally, we consider the nonlinear transmission line network model. This model is used extensively in MOR literature [249, 250, 20, 54], and has become a benchmark problem in MOR. The model includes a chain of resistors and capacitors (cf. Fig. 4.6). A quadratic nonlinearity is introduced at each node by adding nonlinear resistors to the ground given as: $\mathbf{i}_n(\mathbf{x}) = p.\text{sgn}(\mathbf{x})\mathbf{x}^2$ where $\text{sgn}(\mathbf{x}) = 1$ if $\mathbf{x} > 0$ and $\text{sgn}(\mathbf{x}) = -1$ if $\mathbf{x} < 0$. As such, the nonlinear function $\mathbf{f}$ becomes:

$$\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x} - n(\mathbf{x}),$$

where

$$\mathbf{A} = \begin{bmatrix} -2 & 1 & & \\ 1 & -2 & 1 & \\ & \ddots & \ddots & 1 \\ & & 1 & -2 \end{bmatrix}, \quad n(\mathbf{x}) = \begin{bmatrix} \text{sgn}(\mathbf{x}_1)\mathbf{x}_1^2 \\ \text{sgn}(\mathbf{x}_2)\mathbf{x}_2^2 \\ \vdots \\ \text{sgn}(\mathbf{x}_n)\mathbf{x}_n^2 \end{bmatrix},$$

where n is the number of nodes in the circuit. For simplicity, we assume that $p = C = R = 1$. The input to the system is a current source which excites the system, and the output is taken at the first node. The various parameters of the model and reduction are mentioned in Table 4.10).

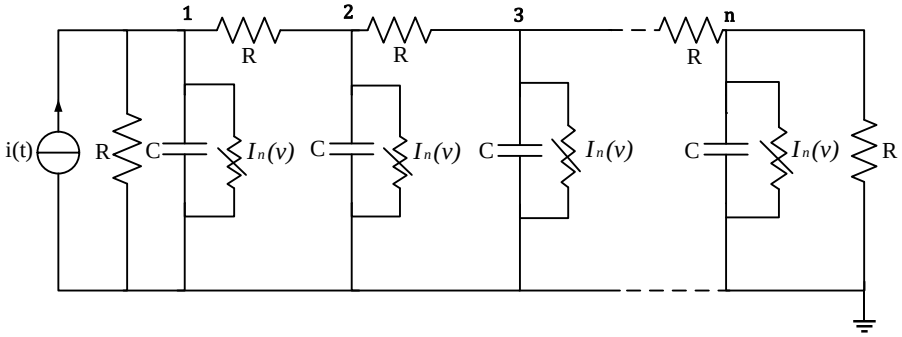

Figure 4.6: Nonlinear transmission line model

Table 4.10: Nonlinear Transmission Line: Parameters used

Method	Parameters
Full model	$n = 1500, R = C = p = 1, t \in [0, 20s]$
NLMM	$r = 25, \dot{\boldsymbol{\zeta}}(t) = 300\cos(300t), i = 1, K = 101, t_k \in [0, 10s]$
DMD	$m_D = 25$
POD	$r = 10, n_s = 300$
DEIM	$m_d = 25$
Test Input	$\mathbf{u}(t) = 15e^{-t}$

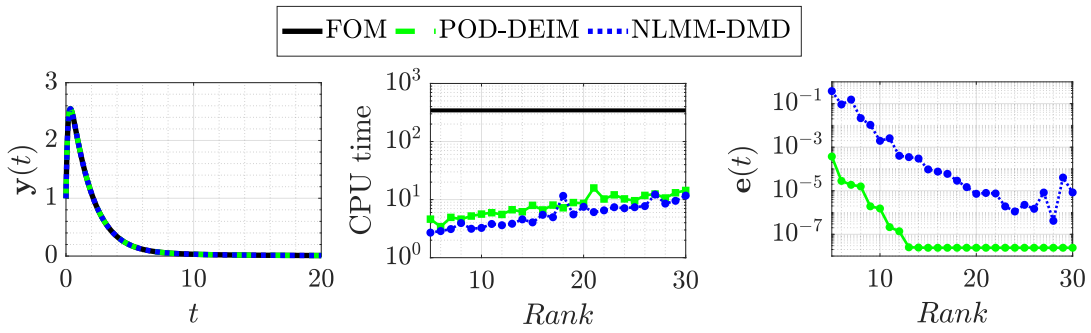

Figure 4.7: Nonlinear Transmission Line: (left) output response, (middle) CPU time (right) relative $\mathcal{L}_2$ norm error

For the application of NLMM, the signal generator used is given as $\dot{\boldsymbol{\zeta}}(t) = 300\cos(300t)$ with $K = 101$ time-snapshots. Simulation results are depicted in Fig. 4.7. As expected, the CPU time of NLMM-DMD is less than POD-DEIM. Since the meaning of rank for DMD and DEIM is different, we also show the error by varying the rank m_d and m_D for a fixed number of POD/NLMM basis functions in Fig. (4.7). We observe that NLMM-DMD always has a monotone decay as we

increase the DMD rank. Finally, Table 4.11 summarizes the comparison of the offline time, online time, and the various error norms for the RC-ladder network. In this case, the application of NLMM-DMD produced around 14 fold reduction in offline time and ≈ 3 fold reduction in online computation time. Thus, a total of 91% savings in overall CPU time was achieved.

Table 4.11: Nonlinear Transmission Line: Numerical results

Method	offline time (s)	online time (s)	$\mathcal{L}_1$	$\mathcal{L}_2$	$\mathcal{L}_\infty$
FOM	-	293.21	-		
POD-DEIM	295.68 (300 NLSE)	11.57	$2.2e^{-5}$	$7e^{-7}$	$3.3e^{-8}$
NLMM-DMD	21.44 (101 NLSE)	3.5	$2.7e^{-4}$	$6e^{-7}$	$2.9e^{-6}$

4.4 Chapter Summary

This chapter discussed two numerical frameworks for the efficient reduction of nonlinear state-space models based on nonlinear moment-matching; NLMM-DEIM and NLMM-DMD. While the former presents a robust MOR approach for obtaining compact ROMs with high accuracy, the latter offers added computational savings in a non-intrusive setup. We presented several numerical simulations to test both the schemes, and the results thus obtained are in compliance with the theoretical observations. In the next chapter, we will extend the application of NLMM for nonlinear models in second-order form.

Chapter 5

Second-Order State-Space Systems

While modeling electrical or mechanical systems, one has to deal with models in second-order form. These include models emerging from structural dynamics, vibroacoustics, MEMS, electrical power systems, and circuits. Furthermore, these models often exhibit material nonlinearities, geometrical nonlinearities (e.g., large deformations), nonlinear boundary conditions, etc. A common approach of reducing such systems is to first transform the second-order structure to the first order and then proceed with the general reduction. However, this results in the loss of the physical meanings of the state variables. Thus, MOR methods that aim to construct reduced models that preserve the second-order structure are strongly aimed [37, 190, 191, 256, 284].

In this chapter, we will focus on reducing a general class of nonlinear second-order state-space models. We will first revisit the reduction for second-order linear systems via moment-matching and the associated approaches. Then, we will discuss the problem in nonlinear settings whereby we will propose another reduction framework based on the second-order NLMM with DEIM (SONLMM-DEIM). Finally, we will discuss the applicability of this framework on some nonlinear, second-order power-system models.

5.1 Moment-matching in linear second-order systems

To define the generalized notion of moments, consider a large-scale, MIMO, linear, second-order state-space model given as:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{D}\dot{\mathbf{x}}(t) + \mathbf{K}\mathbf{x}(t) &= \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t), \end{aligned} \tag{5.1}$$

where the matrices $\mathbf{M}, \mathbf{D}, \mathbf{K} \in \mathbb{R}^{n\times n}$ represent the mass, damping and stiffness matrices respectively. Vector $\mathbf{x}(t) \in \mathbb{R}^n$ represents the displacement vector with $\mathbf{x}(0) = \mathbf{x}_0, \dot{\mathbf{x}}(0) = \dot{\mathbf{x}}_0$, and $\ddot{\mathbf{x}}(0) = \ddot{\mathbf{x}}_0$. Since the mass, stiffness, and damping

matrices in most of the practical systems are symmetric and positive definite, it is often recommended to represent the system (5.1) as a transformed state-space model given as follows:

$$\underbrace{\begin{bmatrix} -\mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{M} \end{bmatrix}}_{\mathbf{E}} \underbrace{\begin{bmatrix} \dot{\mathbf{x}}(t) \\ \ddot{\mathbf{x}}(t) \end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix} \mathbf{0} & -\mathbf{K} \\ -\mathbf{K} & -\mathbf{D} \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \mathbf{x}(t) \\ \dot{\mathbf{x}}(t) \end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix}}_{\mathbf{B}} \mathbf{u}(t), \tag{5.2a}$$

$$\mathbf{y}(t) = \underbrace{\begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix}}_{\mathbf{C}} \underbrace{\begin{bmatrix} \mathbf{x}(t) \\ \dot{\mathbf{x}}(t) \end{bmatrix}}_{\mathbf{x}}. \tag{5.2b}$$

As a result, the symmetry and definiteness of matrix $\mathbf{E}$ is taken care of by matrices $\mathbf{M}$ and $\mathbf{K}$, and the symmetry of matrix $\mathbf{A}$ is maintained by matrices $\mathbf{K}$ and $\mathbf{D}$ respectively.

Definition 5.1 (0-th moment of second-order system) The i-th moment of system (5.1) around expansion point $s = 0$ is defined as:

$$\boldsymbol{\eta}_i^0 = \begin{bmatrix} \mathbf{0} & -\mathbf{C}\mathbf{K}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{0} & -\mathbf{M}\mathbf{K}^{-1} \\ \mathbf{I} & -\mathbf{D}\mathbf{K}^{-1} \end{bmatrix}^i \begin{bmatrix} \mathbf{0} \\ \mathbf{B} \end{bmatrix}. \tag{5.3}$$

▲

It is hereby assumed that the matrix $\mathbf{K}$ is invertible to ensure that the matrix $\mathbf{A}$ remains nonsingular.

Definition 5.2 (Markov parameter of second-order system) The Markov parameter of system (5.1) is defined as:

$$\mathbf{M}_i = \begin{bmatrix} \mathbf{0} & \mathbf{C}\mathbf{M}^{-1} \end{bmatrix} \begin{bmatrix} -\mathbf{D}\mathbf{M}^{-1} & \mathbf{I} \\ -\mathbf{K}\mathbf{M}^{-1} & \mathbf{0} \end{bmatrix}^i \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix}, \tag{5.4}$$

with the condition that matrix $\mathbf{M}$ is invertible. ▲

Definition 5.3 (Second-Order Krylov subspace) *[257]* Given two matrices $\mathbf{X}_1$ and $\mathbf{X}_2 \in \mathbb{R}^{n\times n}$ and another matrix $\mathbf{H}_1 \in \mathbb{R}^{n\times m}$ whose columns represent the starting vectors, the second-order Krylov subspace is defined as:

$$\mathcal{K}_r(\mathbf{X_1}, \mathbf{X}_2, \mathbf{H}_1) = colspan\{\mathbf{Q}_0, \mathbf{Q}_1, ..., \mathbf{Q}_{r-1}\}, \tag{5.5}$$

where

$$\begin{cases} \mathbf{Q}_0 = \mathbf{H}_1, \ \mathbf{Q}_1 = \mathbf{X}_1\mathbf{Q}_0 \\ \mathbf{Q}_i = \mathbf{X}_1\mathbf{Q}_{i-1} + \mathbf{X}_2\mathbf{Q}_{i-2}, \ i = 2, 3, .. \end{cases} \tag{5.6}$$

▲

Now, depending upon the type of damping involved, we distinguish between the following second-order input and output Krylov subspaces.

Definition 5.4 (Second-order input and output Krylov subspace)

- For zero or proportional damping, the columns of $\mathbf{V}$ and $\mathbf{W}$ are chosen as the basis of the following Krylov subspaces given as:

$$\mathcal{K}_r(\mathbf{K}_\sigma^{-1}\mathbf{M}, \mathbf{K}_\sigma^{-1}\mathbf{B}) \subset colspan(\mathbf{V}), \tag{5.7a}$$

$$\mathcal{K}_r(\mathbf{K}_\sigma^{-T}\mathbf{M}^{\mathbf{T}}, \mathbf{K}_\sigma^{-T}\mathbf{C}^{\mathbf{T}}) \subset colspan(\mathbf{W}). \tag{5.7b}$$

- For a general damping ($\mathbf{D} \neq \mathbf{0}$), the columns of $\mathbf{V}$ and $\mathbf{W}$ form a basis of the following second-order Krylov subspaces

$$\mathcal{K}_r(\mathbf{K}_\sigma^{-1}\mathbf{D}_\sigma, \mathbf{K}_\sigma^{-1}\mathbf{M}, \mathbf{K}_\sigma^{-1}\mathbf{B}) \subset colspan(\mathbf{V}), \tag{5.8a}$$

$$\mathcal{K}_r(\mathbf{K}_\sigma^{-T}\mathbf{D}_\sigma^{T}, \mathbf{K}_\sigma^{-T}\mathbf{M}^{\mathbf{T}}, \mathbf{K}_\sigma^{-T}\mathbf{C}^{\mathbf{T}}) \subset colspan(\mathbf{W}), \tag{5.8b}$$

where $\mathbf{K}_\sigma = \mathbf{K} + \sigma\mathbf{D} + \sigma^2\mathbf{M}$ and $\mathbf{D}_\sigma = \mathbf{D} + 2\sigma\mathbf{M}$ ▲

Similar to first-order systems, the reduction techniques employing only one second-order Krylov subspace are known as the *one-sided Krylov subspace method*, and the reduction techniques involving both the second-order Krylov subspaces are known as the *two-sided Krylov subspace methods*.

Now consider the approximation ansatz given as $\mathbf{x}(t) = \mathbf{V}\mathbf{x}_\mathbf{r}(t), \mathbf{V} \in \mathbb{R}^{n\times r}$. Substituting this approximation in (5.1) and pre-multiplying by $\mathbf{W}^T \in \mathbb{R}^{n\times r}$ yields a second-order reduced model of dimension r given as:

$$\begin{aligned} \mathbf{M}_r\ddot{\mathbf{x}}_r(t) + \mathbf{D}_r\dot{\mathbf{x}}_r(t) + \mathbf{K}_r\mathbf{x}_r(t) &= \mathbf{B}_r\mathbf{u}(t), \\ \mathbf{y}_r(t) &= \mathbf{C}_r\mathbf{x}_r(t), \end{aligned} \tag{5.9}$$

where

$$\mathbf{M_r} = \mathbf{W}^T\mathbf{M}\mathbf{V},\ \mathbf{D_r} = \mathbf{W}^T\mathbf{D}\mathbf{V}, \mathbf{K_r} = \mathbf{W}^T\mathbf{K}\mathbf{V}, \mathbf{B_r} = \mathbf{W}^T\mathbf{B}, \mathbf{C_r} = \mathbf{C}\mathbf{V}. \tag{5.10}$$

Using the formulation in (5.2), the second-order reduced model (5.9) is expressed in a transformed state-space given as:

$$\begin{cases} \underbrace{\begin{bmatrix} -\mathbf{W^T}\mathbf{K}\mathbf{V} & \mathbf{0} \\ \mathbf{0} & \mathbf{W}^T\mathbf{M}\mathbf{V} \end{bmatrix}}_{\mathbf{E_r}} \underbrace{\begin{bmatrix} \dot{\mathbf{x}}_\mathbf{r}(t) \\ \ddot{\mathbf{x}}_\mathbf{r}(t) \end{bmatrix}}_{\dot{\mathbf{x}}_\mathbf{r}} = \\ \underbrace{\begin{bmatrix} \mathbf{0} & -\mathbf{W}^T\mathbf{K}\mathbf{V} \\ -\mathbf{W}^T\mathbf{K}\mathbf{V} & -\mathbf{W}^T\mathbf{D}\mathbf{V} \end{bmatrix}}_{\mathbf{A_r}} \underbrace{\begin{bmatrix} \mathbf{x}_\mathbf{r}(t) \\ \dot{\mathbf{x}}_\mathbf{r}(t) \end{bmatrix}}_{\mathbf{x_r}} + \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{W}^T\mathbf{B} \end{bmatrix}}_{\mathbf{B_r}} \mathbf{u}(t), \\ \mathbf{y_r}(t) = \underbrace{\begin{bmatrix} \mathbf{C}\mathbf{V} & \mathbf{0} \end{bmatrix}}_{\mathbf{C_r}} \underbrace{\begin{bmatrix} \mathbf{x}_\mathbf{r}(t) \\ \dot{\mathbf{x}}_\mathbf{r}(t) \end{bmatrix}}_{\mathbf{x_r}}, \end{cases} \tag{5.11}$$

where the projection matrices are defined as:

$$\tilde{\mathbf{V}} = \begin{bmatrix} \mathbf{V} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \end{bmatrix}, \quad \tilde{\mathbf{W}} = \begin{bmatrix} \mathbf{W} & \mathbf{0} \\ \mathbf{0} & \mathbf{W} \end{bmatrix}. \tag{5.12}$$

This formulation preserves the structure of the original model. To obtain the projection basis to match any desired moments, all the previously mentioned theorems in Chapter 2 can be generalized for the second-order system in a similar fashion.

Theorem 5.1 (Moment-matching in linear second-order systems) *[257] In order to match the first $r_1 + r_2$ moments between the original second-order model (5.1) and the reduced second-order model (5.9), it is required that the columns of projection matrices $\mathbf{V}$ and $\mathbf{W}$ used in (5.10) form the basis for the input and output second-order Krylov subspaces (5.8a) and (5.8b) respectively. Also, it is assumed that the matrices $\mathbf{K}$ and $\mathbf{K}_r$ are invertible.* ▲

Similarly the rational interpolation can achieved as follows:

Theorem 5.2 (Rational interpolation in second-order systems) *[257, 256] In order to match first r_i moments about $\sigma_i, i = 1, .., k_2$ of the original and reduced models, it is assumed that matrices $\mathbf{V}$ and $\mathbf{W}$, used in (5.10), are chosen as follows:*

$$\begin{aligned} &span(\mathbf{V}) = \\ &\bigcup_{i=1}^{k_1} \mathcal{K}_{r_i}(-(\mathbf{K} + s_i\mathbf{D} + s_i^2\mathbf{M})^{-1}(\mathbf{D} + 2s_i\mathbf{M}), \\ &-(\mathbf{K} + s_i\mathbf{D} + s_i^2\mathbf{M})^{-1}\mathbf{M}, -(\mathbf{K} + s_i\mathbf{D} + s_i^2\mathbf{M})^{-1}\mathbf{B}), \end{aligned}$$

and

$$\begin{aligned} &span(\mathbf{W}) = \\ &\bigcup_{i=k_1+1}^{k_2} \mathcal{K}_{r_i}(-(\mathbf{K} + s_i\mathbf{D} + s_i^2\mathbf{M})^{-T}(\mathbf{D} + 2s_i\mathbf{M})^T, \\ &-(\mathbf{K} + s_i\mathbf{D} + s_i^2\mathbf{M})^{-T}\mathbf{M}^T, -(\mathbf{K} + s_i\mathbf{D} + s_i^2\mathbf{M})^{-T}\mathbf{C}^T). \end{aligned}$$

▲

The first successful attempt in reducing second-order models came from Meyer and Srinivasan [206] which involved the evaluation of the free-velocity and the zero-velocity gramians. This was later followed by the much improved second-order balancing truncation method by Chahlaoui et al. [75]. However, balancing for second-order systems is not recommended from a numerical perspective. Instead, faster iterative schemes based on Arnoldi or Lanczos are more suitable for reducing systems in the second-order structure. An early attempt in this direction came

from Su, and Craig [284] and later by Bastian and Haase [37] in which Krylov subspace methods were employed. However, these methods could not match nonzero moments, and the number of moments matched was less than the classical Krylov subspace methods. Consequently, several papers appeared based on similar ideas [120, 185]. These methods obtained reduced models by applying different projection mappings to an equivalent state-space model while preserving its structure. Behnam et al. [256] proposed two different techniques for reducing second-order system by first reducing an equivalent state-space model followed by back conversion of the reduced model to second-order form. This resulted in matching double the number of moments than the second-order Krylov methods proposed in Refs. [257, 191, 35].

5.2 Nonlinear second-order systems

Now, we consider a nonlinear second-order state-space model given as:

$$\begin{aligned} \mathbf{M}\ddot{\mathbf{x}}(t) + \mathbf{D}\dot{\mathbf{x}}(t) + \mathbf{f}(\mathbf{x}(t)) &= \mathbf{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t), \end{aligned} \tag{5.13}$$

where $\mathbf{M} \in \mathbb{R}^{n\times n}$ is a non-singular mass matrix, $\mathbf{D} \in \mathbb{R}^{n\times n}$ represents the damping term. For nonlinear systems, a linear Rayleigh damping ($\mathbf{D} = \boldsymbol{\alpha}\mathbf{M} + \boldsymbol{\beta}\mathbf{K}(\mathbf{x}_{\text{eq}})$) or zero damping ($\mathbf{D} = \mathbf{0}$) is mostly considered. $\mathbf{f}(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}^n$ is the nonlinear smooth mapping. For the equilibrium point $\mathbf{x}_{\text{eq}}$ to be exponentially stable, it is hereby assumed that the mass matrix is symmetric positive definite, i.e., $\mathbf{M} = \mathbf{M}^T \succ \mathbf{0}$.

5.2.1 Linear Galerkin projection

The most straight forward approach to reduce nonlinear second-order systems is to employ a linear Petrov-Galerkin projection. Substituting the linear approximation ansatz (cf. 2.34) in FOM (5.1), and premultiplying the overdetermined system with the projector $\mathbf{P} = \mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T$ gives:

$$\mathbf{P}\left(\mathbf{M}\mathbf{V}\ddot{\mathbf{x}}_r(t) + \mathbf{D}\mathbf{V}\dot{\mathbf{x}}_r(t) + \mathbf{f}(\mathbf{V}\mathbf{x}_r(t)) - \mathbf{B}\mathbf{u}(t) - \mathbf{r}(t)\right) = \mathbf{0}. \tag{5.14}$$

And enforcing the *Petro-Galerkin* condition vanishes the residual and yields the desired ROM:

$$\begin{aligned} \mathbf{M}_r\ddot{\mathbf{x}}_r(t) + \mathbf{D}_r\dot{\mathbf{x}}_r(t) + \mathbf{V}^T\mathbf{f}(\mathbf{V}\mathbf{x}_r(t)) &= \mathbf{B}_r\mathbf{u}(t), \\ \mathbf{y}_r(t) &= \mathbf{C}_r\mathbf{x}_r(t), \end{aligned} \tag{5.15}$$

where $\{\mathbf{M_r}, \mathbf{D_r}\} = \mathbf{W^T}\{\mathbf{M}, \mathbf{D}\}\mathbf{V}, \mathbf{B_r} = \mathbf{W^T}\mathbf{B}, \mathbf{C_r} = \mathbf{CV}$ with $\mathbf{W} = \mathbf{V}$, and $\{\mathbf{x_r}(0), \dot{\mathbf{x}}_\mathbf{r}(0)\} = (\mathbf{W^T V})^{-1}\mathbf{W^T}\{\mathbf{x_0}, \dot{\mathbf{x}}_\mathbf{0}\}$.

5.2.2 Nonlinear Galerkin projection

Another possibility of obtaining a reduced second-order model is by applying a *nonlinear Petrov-Galerkin* projection (cf. 3.2) given as:

$$\mathbf{x}(t) = \wp(\mathbf{x}_r(t)) + \mathbf{e}(t), \tag{5.16}$$

where the first and second derivative yields:

$$\dot{\mathbf{x}}(t) = \underbrace{\frac{\partial \wp(\mathbf{x}_r(t))}{\partial \mathbf{x}_r(t)}}_{\tilde{\mathbf{V}}_{\mathbf{x}_r}} \dot{\mathbf{x}}_\mathbf{r}(t) + \dot{\mathbf{e}}(t), \tag{5.17a}$$

$$\ddot{\mathbf{x}}(t) = \underbrace{\frac{\partial \wp(\mathbf{x}_r(t))}{\partial \mathbf{x}_r(t)}}_{\tilde{\mathbf{V}}_{\mathbf{x}_r}} \ddot{\mathbf{x}}_\mathbf{r}(t) + \underbrace{\frac{\partial^2 \wp(\mathbf{x}_r(t))}{\partial \mathbf{x}_r^2(t)}}_{\mathbf{d}\tilde{\mathbf{V}}_{\mathbf{x}_r}} (\dot{\mathbf{x}}_\mathbf{r}(t) \otimes \dot{\mathbf{x}}_\mathbf{r}(t)) + \ddot{\mathbf{e}}(t), \tag{5.17b}$$

and where the matrices

$$\tilde{\mathbf{V}}_{\mathbf{x}_r} = \frac{\partial \wp(\mathbf{x}_r(t))}{\partial \mathbf{x}_r(t)} \in \mathbb{R}^{n \times r}, \qquad \mathbf{d}\tilde{\mathbf{V}}_{\mathbf{x}_r} = \frac{\partial^2 \wp(\mathbf{x}_r(t))}{\partial \mathbf{x}_r^2(t)} \in \mathbb{R}^{n \times r^2} \tag{5.18}$$

are the Jacobian and Hessian matrices respectively. Substituting the the approximation ansatz (5.16) and its derivatives (5.17b) and (5.17b) yields an overdetermined system of equations with residual $\mathbf{r}(t) \in \mathbb{R}^n$. The resulting system is then projected onto an orthogonal tangent space $\mathcal{V}^\perp = \text{ran}(\tilde{\mathbf{V}}_{\mathbf{x}_r})$ using the projector $\tilde{\mathbf{P}} = \tilde{\mathbf{V}}_{\mathbf{x}_r}(\tilde{\mathbf{V}}_{\mathbf{x}_r}^T \tilde{\mathbf{V}}_{\mathbf{x}_r})^{-1} \tilde{\mathbf{V}}_{\mathbf{x}_r}^T$ which yields

$$\tilde{\mathbf{P}} \left(\mathbf{M}\tilde{\mathbf{V}}_{\mathbf{x}_r} \ddot{\mathbf{x}}_r + \mathbf{M}\mathbf{d}\tilde{\mathbf{V}}_{\mathbf{x}_r} (\dot{\mathbf{x}}_r \otimes \dot{\mathbf{x}}_r) + \mathbf{D}\tilde{\mathbf{V}}_{\mathbf{x}_r} \dot{\mathbf{x}}_r + \mathbf{f}(\wp(\mathbf{x}_r)) - \mathbf{B}\mathbf{u}(t) - \mathbf{r}(t) \right) = \mathbf{0} \tag{5.19}$$

Finally enforcing the Galerkin condition $\tilde{\mathbf{V}}_{\mathbf{x}_r}^T \mathbf{r}(t) = \mathbf{0}$ yields the projected ROM:

$$\begin{aligned} \tilde{\mathbf{M}}_r \ddot{\mathbf{x}}_r(t) + \tilde{\mathbf{p}}_\mathbf{r}(t) + \tilde{\mathbf{D}}_r \dot{\mathbf{x}}_r(t) + \mathbf{V}^T \mathbf{f}(\wp(\mathbf{x}_r(t))) &= \tilde{\mathbf{B}}_r \mathbf{u}(t), \\ \mathbf{y}_r(t) &= \mathbf{C}\wp(\mathbf{x}_r(t)), \end{aligned} \tag{5.20}$$

where $\{\tilde{\mathbf{M}}_r, \tilde{\mathbf{D}}_r\} = \tilde{\mathbf{V}}_{\mathbf{x}_r}^T \{\mathbf{M}, \mathbf{D}\} \tilde{\mathbf{V}}_{\mathbf{x}_r}, \tilde{\mathbf{p}}_r(t) = \tilde{\mathbf{V}}_{\mathbf{x}_r}^T \mathbf{M}\mathbf{d}\tilde{\mathbf{V}}_{\mathbf{x}_r} (\dot{\mathbf{x}}_r \otimes \dot{\mathbf{x}}_r)$, and $\tilde{\mathbf{B}} = \tilde{\mathbf{V}}_{\mathbf{x}_r}^T \mathbf{B}$.

5.2.3 Moment-matching in nonlinear second-order systems

Consider a nonlinear, second-order exogenous system/ signal-generator as follows:

$$\begin{aligned}\dot{\boldsymbol{\zeta}}(t) &= \boldsymbol{\varpi}(\boldsymbol{\zeta}(t)),\\ \ddot{\boldsymbol{\zeta}}(t) &= \frac{\partial \boldsymbol{\varpi}(\boldsymbol{\zeta}(t)))}{\partial \boldsymbol{\zeta}(t)}.\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)),\\ \boldsymbol{u}(t) &= \boldsymbol{\varrho}(\boldsymbol{\zeta}(t)),\end{aligned} \tag{5.21}$$

with smooth mappings $\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) : \mathbb{R}^r \to \mathbb{R}^r$ and $\boldsymbol{\varrho}(\boldsymbol{\zeta}(t)) : \mathbb{R}^r \to \mathbb{R}^m$. It is assumed that the the nonlinear signal generator 5.21 is neutrally stable and observable.

Theorem 5.3 (Nonlinear moment-matching for second-order systems) *[26] With the assumption that the nonlinear second-order signal generator (5.21) is neutrally-stable, the nonlinear second-order ROM (5.20) matches the (well-defined) steady-state of FOM (5.13) when both are excited by the inputs from (5.21). The projection mapping $\boldsymbol{\wp}(\boldsymbol{\zeta}(t))$ used to obtain the ROM uniquely solves the second-order Sylvester PDE given as:*

$$\begin{aligned}\mathbf{M}\frac{\partial \boldsymbol{\wp}(\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}}\frac{\partial \boldsymbol{\varpi}(\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}}\boldsymbol{\varpi}(\boldsymbol{\zeta}) + \mathbf{M}\frac{\partial^2 \boldsymbol{\wp}(\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}^2}(\boldsymbol{\varpi}(\boldsymbol{\zeta}) \otimes \boldsymbol{\varpi}(\boldsymbol{\zeta})) +\\ \mathbf{D}\frac{\partial \boldsymbol{\wp}(\boldsymbol{\zeta})}{\partial \boldsymbol{\zeta}}\boldsymbol{\varpi}(\boldsymbol{\zeta}) + \mathbf{f}(\boldsymbol{\wp}(\boldsymbol{\zeta})) = \mathbf{B}\boldsymbol{\varrho}(\boldsymbol{\zeta}).\end{aligned} \tag{5.22}$$

▲

The steady-state response is related to the 0^{th} nonlinear moments at $(\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t)))$, and is given as:

$$\mathbf{y}_{\text{ss}} = \boldsymbol{\eta}_0(\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t))). \tag{5.23}$$

Remark 5.1 (Interpretation as steady-state matching) Interconnecting the second-order signal generator (5.21) with the FOM (5.13) and ROM (5.20) can be interpreted as exact matching of the steady-state of the FOM (5.13) and ROM (5.20) (cf. Fig. 5.1), i.e.,

$$\mathbf{y}_{\text{ss}}(t) = \boldsymbol{\eta}_0(\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t))) = \boldsymbol{\eta}_{r0}(\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)), \boldsymbol{\varrho}(\boldsymbol{\zeta}(t))) = \mathbf{y}_{\text{r,ss}}(t) \tag{5.24}$$

5.3 Approximated second-order nonlinear moment-matching

The solution of second-order nonlineatr Sylvester PDE (5.22) is expensive to solve for $\boldsymbol{\varpi}(\boldsymbol{\zeta}(t))$ as it involves solving a PDE in second-order structure. As such,

we take some simplifications from Maria et al. [295] to arrive at a practical, *simulation-free* algorithm. These simplifications are much similar to what was used for the first-order Sylvester PDE as discussed in Section 4.1.1. Here, we only discuss the a general case of a nonlinear signal generator.

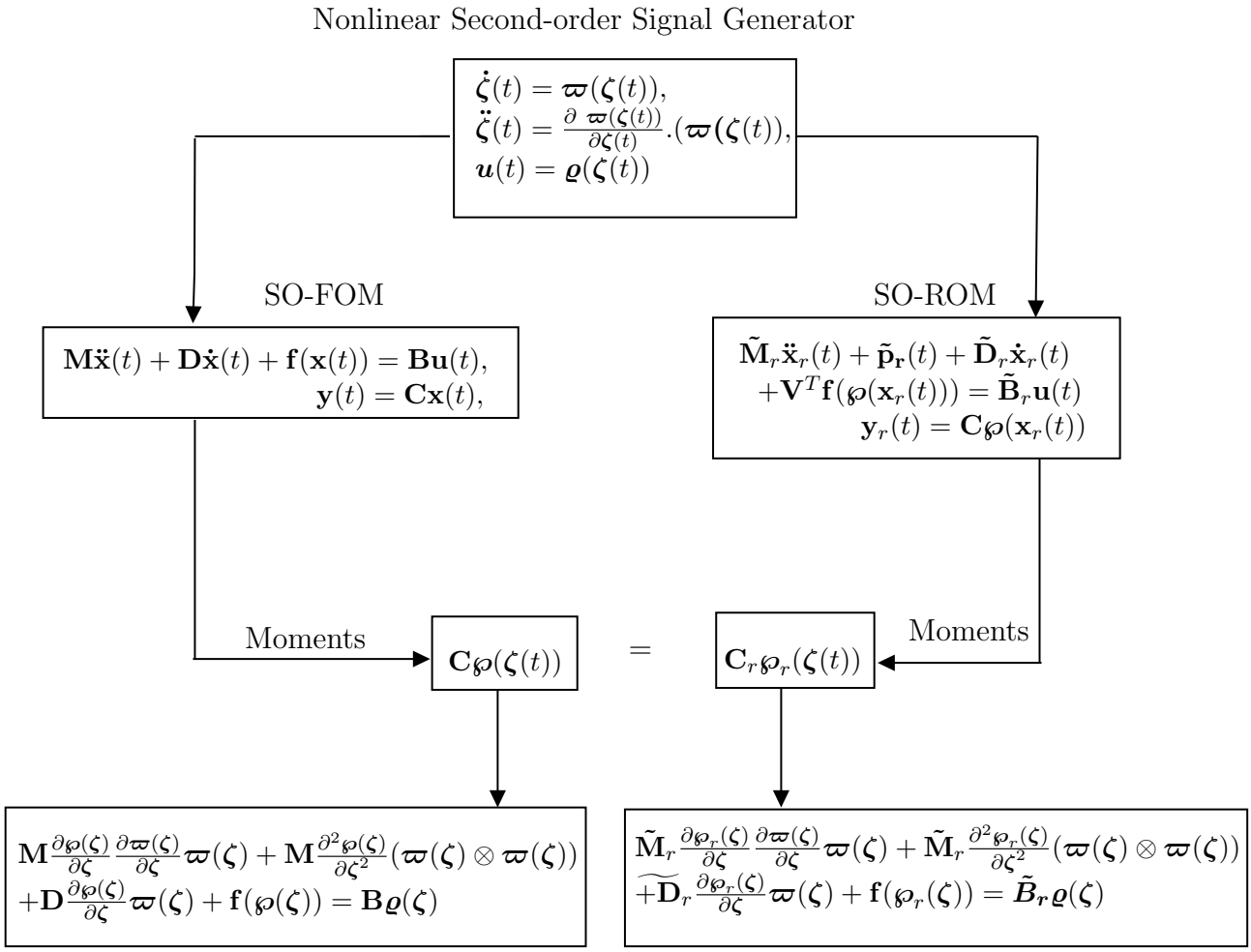

Figure 5.1: Time-domain illustration of nonlinear second-order moment-matching in terms of steady-state response matching between SO-FOM and SO-ROM

1) **Linear projection**
Using a linear projection $\mathbf{x}(t) = \mathbf{V}\boldsymbol{\zeta}(t)$, the second-order Sylvester PDE (5.22) becomes:

$$\mathbf{M}\mathbf{V}\frac{\partial\boldsymbol{\varpi}(\boldsymbol{\zeta}(t))}{\partial\boldsymbol{\zeta}(t)}\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) + \mathbf{D}\mathbf{V}\boldsymbol{\varpi}(\boldsymbol{\zeta}(t)) + \mathbf{f}(\mathbf{V}\boldsymbol{\zeta}(t)) - \mathbf{B}\boldsymbol{\varrho}(\boldsymbol{\zeta}(t)) = \mathbf{0}. \quad (5.25)$$

Note that the application of linear projection vanishes the term corresponding to second-order partial derivative of $\boldsymbol{\varpi}(\boldsymbol{\zeta}(t))$.

2) Column-wise consideration
System (5.25) is then considered column-wise for each $\mathbf{v}_i$ $i = 1, .., r$, to avoid the underdetermined problem of n equations for $n.r$ unknowns in $\mathbf{V} \in \mathbb{R}^{n \times r}$ which yields

$$\mathbf{M}\mathbf{v}_i \frac{\partial \boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t))}{\partial \boldsymbol{\zeta}_i(t)} \boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t)) + \mathbf{D}\mathbf{v}_i \boldsymbol{\varpi}_i(\boldsymbol{\zeta}_i(t)) + \mathbf{f}(\mathbf{v}_i \boldsymbol{\zeta}_i(t)) - \mathbf{B}\boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t)) = \mathbf{0}. \tag{5.26}$$

3) Time discretization
Finally, (5.26) is solved for time-discrete samples $t_k, k = 1, .., K$ yielding:

$$\mathbf{M}\mathbf{v}_{ik} \frac{\partial \boldsymbol{\varpi}_{ik}(\boldsymbol{\zeta}_{ik}(t_k))}{\partial \boldsymbol{\zeta}_{ik}(t_k)} \boldsymbol{\varpi}_{ik}(\boldsymbol{\zeta}_{ik}(t_k)) + \mathbf{D}\mathbf{v}_{ik} \boldsymbol{\varpi}_{ik}(\boldsymbol{\zeta}_{ik}(t_k)) + \mathbf{f}(\mathbf{v}_{ik} \boldsymbol{\zeta}_{ik}(t_k)) - \mathbf{B}\boldsymbol{\varrho}_{ik}(\boldsymbol{\zeta}_{ik}(t_k)) = \mathbf{0}. \tag{5.27}$$

Note that prior to solving (5.27), the discrete solution of $\boldsymbol{\zeta}_{ik}(t_k)$ of the nonlinear second-order signal generator (5.21) should be computed.

Thus, these simplifications result in the following second-order nonlinear moment-matching algorithm:

Algorithm 8 Second-Order Nonlinear Moment Matching (SONLMM)

INPUT: $\mathbf{M}, \mathbf{D}, \mathbf{B}, \mathbf{f}(\mathbf{x}, \mathbf{u}), \mathbf{K}(\mathbf{x}, \mathbf{u}), \boldsymbol{\zeta}_i(t_k), \dot{\boldsymbol{\zeta}}_i(t_k), \ddot{\boldsymbol{\zeta}}_i(t_k), \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t_k))$, initial guess $\mathbf{v}_{0,ik}$, reduced rank order r
OUTPUT: Orthogonal basis: $\mathbf{V}$

```
for i = 1 : r do
    for k = 1 : K do
        fun = @(v) Mv ζ̈_i(t_k) + Dv ζ̇_i(t_k) + f(v ζ_i(t_k)) − B ϱ_i(ζ_i(t_k))   ▷ residual
        Jfun = @(v) M ζ̈_i(t_k) + D ζ̇_i(t_k) + K(v ζ_i(t_k)) ζ_i(t_k)   ▷ Jacobian of residual
        v_ik = NewtonRaphson(fun, v_0,ik, Jfun)
        V(:, (i − 1) ∗ K + k) ← v_ik
        V = GramSchmidt(v_ik, V)        ▷ optional
    end for
end for
[U, Σ, ˜] = SVD(V);   V = U(:, 1 : r)        ▷ optional
```

5.4 SONLMM-DEIM

Now we are ready to introduce another efficient MOR strategy to treat nonlinear systems in the second-order form. The method is called SONLMM-DEIM.

The motivation of using DEIM with SONLMM stems from the fact that the complexity in solving reduced nonlinear function $\mathbf{f}()$ in ROMs (5.15) and (5.20) still depends on the original model dimension n, which is expensive. Thus, employing a hyper-reduction-based scheme such as DEIM can reduce the complexity of solving nonlinear inner products to a cost proportional to the number of reduced variables. Using the DEIM approximation of the nonlinear function $\mathbf{f}()$ the ROM (5.15) becomes:

$$\begin{aligned} \mathbf{M}_r\ddot{\mathbf{x}}_r(t) + \mathbf{D}_r\dot{\mathbf{x}}_r(t) + \boldsymbol{\beta}_{\text{DEIM}}\mathbf{f}(\boldsymbol{\gamma}_{\text{DEIM}}\mathbf{x}_r(t)) &= \mathbf{B}_r\mathbf{u}(t), \\ \mathbf{y}_r(t) &= \mathbf{C}_r\mathbf{x}_r(t), \end{aligned} \tag{5.28}$$

where the terms $\boldsymbol{\beta}_{\text{DEIM}} = \mathbf{W}^T\mathbf{U}_\mathbf{d}(\mathbf{P}^T\mathbf{U}_\mathbf{d})^{-1} \in \mathbb{R}^{r\times m_d}$ and $\boldsymbol{\gamma}_{\text{DEIM}} = \mathbf{P}^T\mathbf{V} \in \mathbb{R}^{m_d\times r}$ are both independent of n and hence can be precomputed. The algorithm is given as follows:

Algorithm 9 SONLMM-DEIM

Input: $\mathbf{M}, \mathbf{D}, \mathbf{B}, \mathbf{f}(\mathbf{x},\mathbf{u}), \mathbf{K}(\mathbf{x},\mathbf{u}), \boldsymbol{\zeta}_i(t_k), \dot{\boldsymbol{\zeta}}_i(t_k), \ddot{\boldsymbol{\zeta}}_i(t_k), \boldsymbol{\varrho}_i(\boldsymbol{\zeta}_i(t_k))$, initial guess $\mathbf{v}_{0,ik}$, reduced rank order r, and m_d number of DEIM modes.

1: Compute $\mathbf{V}$ following Algorithm 8 ▷ orthogonal basis via SONLMM
2: Obtain function snapshots $\mathbf{F} = [\mathbf{f}(\mathbf{x}(t_1))\ \mathbf{f}(\mathbf{x}(t_2))...\ \mathbf{f}(\mathbf{x}(t_{n_s}))]$
3: Calculate $[\mathbf{u}_l,\ \boldsymbol{\Sigma}_\mathbf{d}, \tilde{}] = \mathbf{SVD}(\mathbf{F})$
4: Obtain $\mathbf{U}_\mathbf{d}, \mathbf{P}$ and $\vec{p}$ from Algorithm 3 ▷ DEIM basis
5: Construct the reduced model (5.28) with the reduced nonlinear function obtained from (3.47)
6: Integrate (5.28)
7: Project back full solution

Since the discussion regarding the computational aspects also applies here, we refrain from the further details and instead refer the reader to Section 4.1.2. In the following, we will discuss the MOR applications in power systems which are primarily second-order in nature. We will first show a general reduction approach by converting the second-order structure of the power system models to a first-order. Afterward, we will demonstrate the use of SONLMM-DEIM to obtain a structure-preserving ROM.

5.5 Reduction of Power System models

Due to the precise nature of the electrical power grids spanning a wide geographical area, their mathematical models may easily have dimensions of several thousand degrees of freedom. Several applications like security assessment, dynamic simulation, online system identification, trajectory sensitivity analysis, and control of power systems require repeated simulations with higher precision [287]. As such,

these large-scale models pose an onerous challenge in terms of computational resources, e.g., core hours. In power system parlance, the model reduction problem is also known as the problem of *dynamic equivalencing* [86]. Most often, power systems are divided into a *study area*, which contains the variables of interest, and an *external area* making up the rest of the power system. The study area is modeled in full detail and is precisely analyzed and controlled, while the external area, being a less faithful representation, is usually reduced to a linear system without affecting the behavior of the study area [78]. Based on the physical properties of power systems, coherency-based MOR methods were initially proposed, see, e.g., [290, 128, 289]. These methods first aim to identify a coherent group of generators, i.e., machines with a similar behavior when the same input is applied. Then, the dynamic reduction is performed by replacing such machines with aggregate generators and networks. However, the performance of these methods is limited to the identification of coherent generators. Besides the coherency based methods, techniques based on *synchrony* [244, 211], singular perturbation [307], modal analysis [199, 198] have been proposed. In the past years, there has been an interest in the power system community to develop reduction techniques that are based on mathematical properties of power systems instead of physical ones, see, e.g., [210, 205, 68]. One of the main reasons for this shift is the possibility of reducing networks with renewable sources. These reduction techniques include the classical *Krylov* subspace reduction [78, 262], balanced truncation [245] and extended balanced truncation [280] have successfully been implemented. Moreover, methods such as measurement-based reduction [77, 302], ANN-based boundary matching technique [194], heuristic optimization based approach [73], independent component analysis approach [21] have also been reported. Since most of the available MOR techniques for power systems rely on linearizing the external area, these methods cannot always provide an accurate description of the full model, which is intrinsically nonlinear. Towards this direction, nonlinear MOR techniques for power systems have been recently proposed, see. e.g., [313, 196, 222, 235]. This is also complemented with MOR schemes that preserve certain system parameters as well[1, 262]. However, most of the nonlinear MOR methods either involve expensive measurements of large-scale snapshot ensemble in the presence of some excitation input signal or the solution of *Lyanupov* equations, which are also computationally demanding. The problem is further compounded when repeated simulations are required for different input signals or in parametric systems.

5.5.1 Dynamical Power Grid models

The most critical stability concern for the normal operation of a power system is the synchronization of its oscillators, which may be generators or loads, in terms of their frequencies. The following equation of motion describes the rotational dynamics of i-th alternator a power system with n oscillators:

$$\frac{2\mathbf{H}_i}{\boldsymbol{\omega}_R}\frac{d^2\boldsymbol{\delta}_i}{dt^2} + \mathbf{D}_i\frac{d\boldsymbol{\delta}_i}{dt} = \mathbf{F}_i + \mathbf{f}_i(\boldsymbol{\delta}), \qquad \forall i = 1, 2, .., n \tag{5.29}$$

where $\mathbf{H}_i, \mathbf{D}_i \in \mathbb{R}^{n\times n}$ are the diagonal matrices containing normalized inertia constant and damping factors, $\boldsymbol{\omega}_R$ is the reference frequency, $\boldsymbol{\delta}_i \in \mathbb{R}^n$ represent the rotor angles which are the states of the nonlinear system, and n is the number of generators of the system. $\mathbf{F}_i$ is a constant term representing the input mechanical power while $\mathbf{f}_i(\boldsymbol{\delta})$ is the nonlinear coupling term representing the demanded electrical power of i-th generator described as:

$$\mathbf{f}_i(\boldsymbol{\delta}) = -\sum_{j=1;j\neq i}^{n} \mathbf{K}_{ij}\sin(\boldsymbol{\delta}_i - \boldsymbol{\delta}_j - \boldsymbol{\gamma}_{ij}). \tag{5.30}$$

The constant terms of $\mathbf{K}_{ij}$, $\boldsymbol{\gamma}_{ij}$, and $\mathbf{F}_i$ depend on the solution of power flow equations. It is hereby assumed that under equilibrium, $\mathbf{F}_i = \mathbf{f}_i(\boldsymbol{\delta})$. Whenever there is any variation between the two, their difference is compensated by either decreasing or increasing the rotor's angular momentum, which results in the transient phenomenon. Thus, the transient behavior of the power grid network is given by the solution of (5.29) under the assumption that all nodes are considered as PV nodes. However, this condition is relaxed for a general setup containing both generator and load buses resulting in the following generalized form:

$$\begin{aligned}\alpha_i\Big(\frac{2\mathbf{H}_i}{\boldsymbol{\omega}_R}\frac{d^2\boldsymbol{\delta}_i}{dt^2} + \mathbf{D}_i\frac{d\boldsymbol{\delta}_i}{dt} - \mathbf{F}_i\Big) + (1+\alpha_i)p_i^k \\ = -\sum_{j=1;j\neq i}^{n} \mathbf{K}_{ij}\sin(\boldsymbol{\delta}_i - \boldsymbol{\delta}_j - \boldsymbol{\gamma}_{ij}),\end{aligned} \tag{5.31}$$

where

$$\alpha_i = \begin{cases} 1 : \text{Generator bus} \\ 0 : \text{elsewhere} \end{cases}$$

and p_i^k is power consumed at i^{th} node. Thus, the overall dynamics is governed by (5.31), which is an n-dimensional system of second-order differential equations corresponding to generator buses, and algebraic equations associated with load buses.

Although the swing equation for a single oscillator is simple, modeling a realistic power system is challenging due to the large-scale structure of a typical power network that may even span multiple countries. Furthermore, modeling of loads requires the incorporation of their dynamical behavior at different nodes in the network. To address these problems, different modeling techniques have been proposed. Effective network (EN) and synchronous motor (SM) models are of particular interest as they are most commonly used [218]. Both of these are described by (5.29), albeit with different approaches to the interpretation of $\mathbf{F}_i$, $\mathbf{K}_{ij}$, and $\boldsymbol{\gamma}_{ij}$. The EN model is a network-reduced model in which the number of nodes is reduced from n to n_g, where n_g represents the number of generators, and the network parameters are described as follows:

$$\mathbf{F}_i = \mathbf{P}_{gi} - |\mathbf{E}|^2\,\mathbf{G}_{ij},\ \mathbf{K}_{ij} = \left|\mathbf{E}_i\mathbf{E}_j\mathbf{Y}_{ij}\right|,\ \boldsymbol{\gamma}_{ij} = \boldsymbol{\lambda}_{ij} - \frac{\pi}{2}, \tag{5.32}$$

where $\mathbf{P}_{gi}$ is the steady-state electrical power output of ith generator. $\mathbf{Y}_{ij}$ is the corresponding element of the admittance matrix between the generator terminal nodes obtained by eliminating the load nodes through *Kron* reduction, $\mathbf{G}_{ij}$ is its real part and $\boldsymbol{\lambda}_{ij}$ is its argument. The admittance network may not have a complete graph topology due to the reduction. $\mathbf{E}$ is the voltage at the internal node of the generator behind the transient reactance.

The SM model, on the other hand, represents each load as a synchronous motor or as a generator with negative mechanical power. Thus, the admittance network has the topology of a complete graph as all generators and motors (loads) are accounted for. The parameters are defined by the same equation (5.32) of the EN model, although with a different meaning of the symbols. The internal voltage $\mathbf{E}_i$ is determined for all generators and motors, so is $\mathbf{F}_i$. The SM model has been used to study the network topology of the European power grids and minimum coupling strength required for synchronization [192].

Next, we consider both the EN and SM models after transformation to first-order systems as follows:

$$\dot{\boldsymbol{\delta}}_i = \boldsymbol{\omega}_i, \tag{5.33a}$$

$$\boldsymbol{\omega}_i = -\frac{\mathbf{D}_i}{2\mathbf{H}_i}\boldsymbol{\omega}_i + \frac{\boldsymbol{\omega}_R}{2\mathbf{H}_i}\mathbf{F}_i + \frac{\boldsymbol{\omega}_R}{2\mathbf{H}_i}\mathbf{f}_i(\boldsymbol{\delta}), \tag{5.33b}$$

where $\boldsymbol{\omega}_i$ is the frequency at ith node. We used the IEEE 118 and IEEE 300 bus systems as our test models. All the parameters of the IEEE 118 and 300 bus system models were generated using MATLAB toolbox MATPOWER 6.0 [315], and pg_sync_models [218][1]. The number of oscillators and size of original and reduced models along-with the CPU times for the two bus models are enlisted in Table 5.1

Table 5.1: Different parameters of original and reduced models for the two IEEE bus systems

Bus system	Config.	No. of oscillators	FOM Size & CPU time	ROM Size & CPU time
IEEE-118	EN	54	108 (24.3s)	20 (12.53s)
	SM	118	236 (52.8s)	20 (20.98s)
IEEE-300	EN	69	138 (37.56s)	25 (17.38s)
	SM	300	600 (995.5s)	25 (280.2s)

The reduction of (5.33 was performed using NLMM-DMD described in Sec. 4.3. NLMM was used via Algorithm 4 by selecting the training signal thorough a single

[1]The MATLAB script to reproduce the results are available at `https://zenodo.org/record/4954937#.YMh0OzaA79E`

signal generator selected as $\dot{\boldsymbol{\zeta}}(t) = 5\sin(2\pi t)$ with $K = 40$ and $t_k \in [0\ 20]$. This choice of the signal generator was expected to capture both the transient and steady-state behavior of the system. For the application of DMD, Algorithm 6 was employed, and the nonlinearity was approximated with $m_D = 25$ modes.

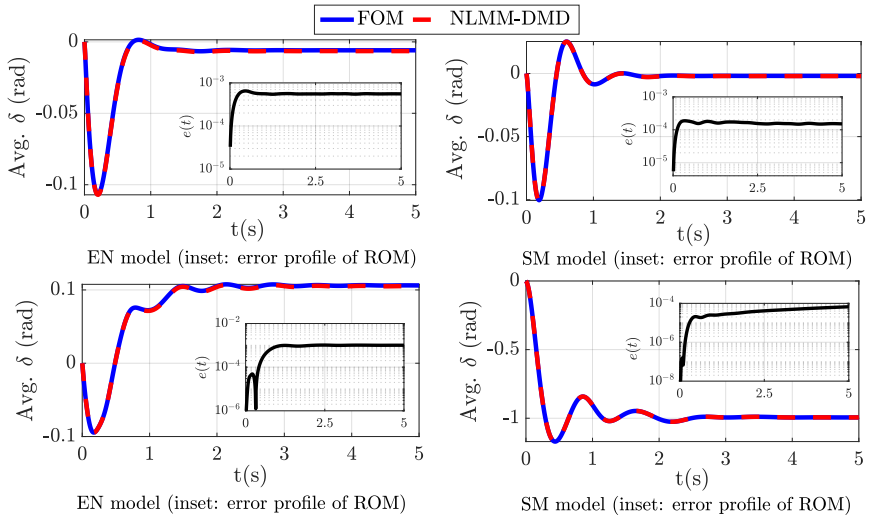

Figure 5.2: Top row: Output response for the full and reduced models of the IEEE 118 bus system (inset: error profile of NLMM-DMD ROM), Bottom row: Output response for the full and reduced models of the IEEE 300 bus system (inset: error profile of NLMM-DMD ROM)

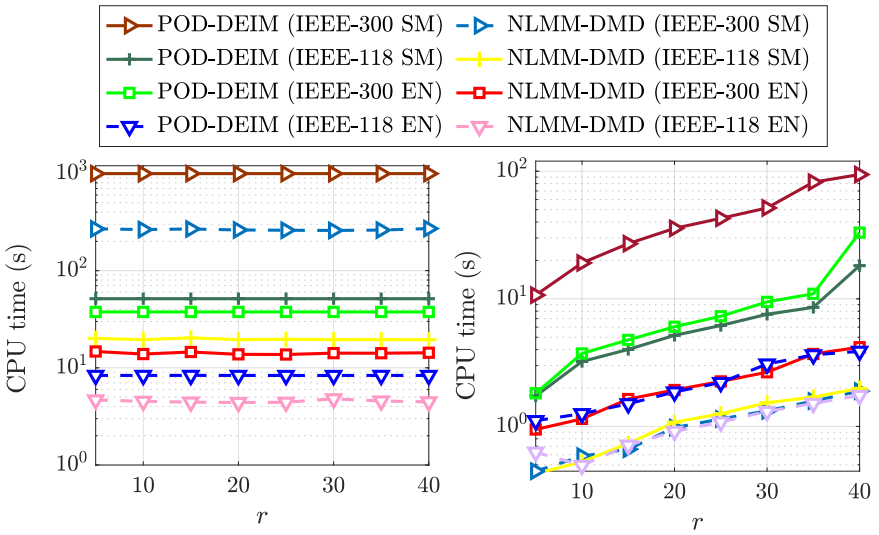

Figure 5.3: Comparison of the simulation times between NLMM-DMD method and POD-DEIM for two IEEE bus systems; (left) offline time (right) online time

The output responses (average of rotor angle ($\boldsymbol{\delta}$)) of the original and NLMM-DMD reduced models are presented in Fig. 5.2 along with point-wise error. It is seen that the reduced model captures the dynamics of the original nonlinear model with high accuracy. A quantitative comparison of NLMM-DMD framework with POD-DEIM is depicted in Fig. 5.3. As expected, the offline CPU times of NLMM are always less than POD for different sizes of ROMs. Similar is true for the online evaluation of ROMs via DMD method over the DEIM.

To further test the validity of the reduced models, we considered a more practical scenario, i.e., we tested the performance of the ROMs for different fault scenarios of the SM model of the IEEE 118 bus system (cf. Fig 5.4) which are discussed as follows.

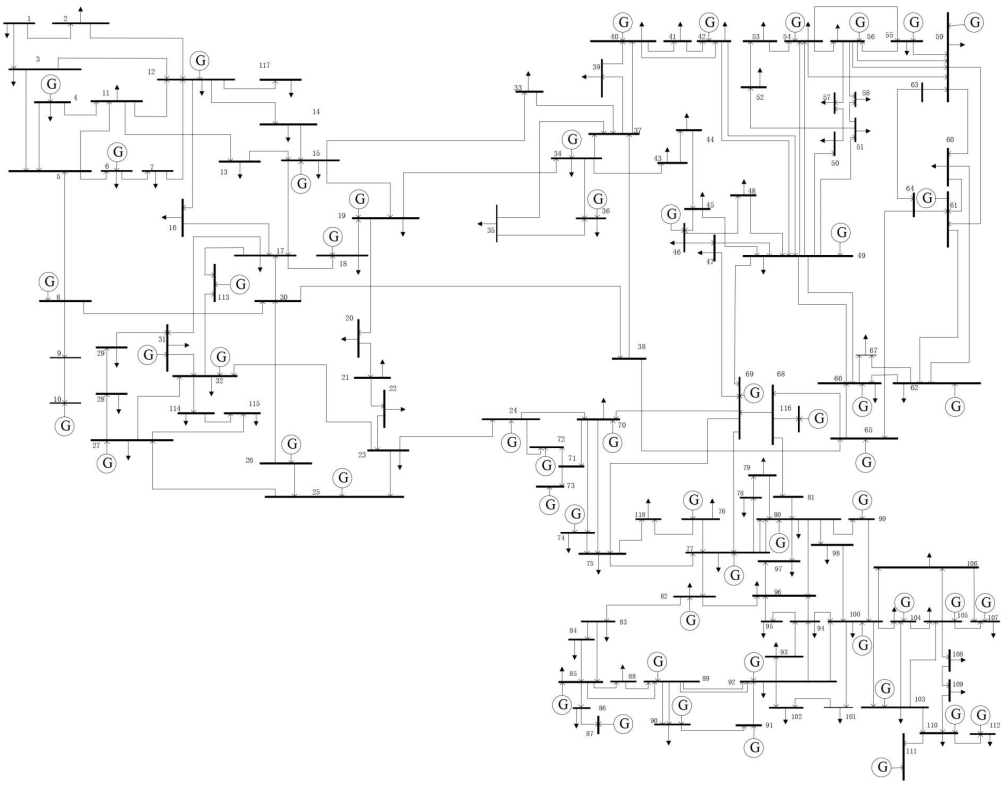

Figure 5.4: Single line diagram of IEEE 118 bus system

(i) **Case I: A generator outage**
In the first scenario, we simulated a generator outage connected at 89^{th} bus (cf. Fig. 5.4). The fault was induced at time $t = 1s$ and the system was able to recover after disturbances. The average velocities (large) of full-order and reduced-order system are shown in Fig. 5.5 along with the error profile. It is seen that the reduced model faithfully captures the transient as well as the steady-state profile of the original response with a high approximation. A comparison of the elapsed CPU times with POD-DEIM along-with $\mathcal{L}_2$ error is presented in Table 5.2.

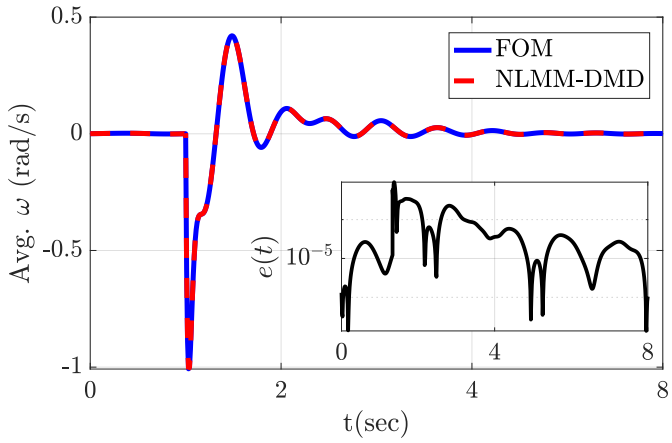

Figure 5.5: First Scenario: Average angular velocities of the full-order and reduced system of the IEEE 118 bus system. Inset: absolute error of the reduced model

(ii) **Case-II: A self-clearing 3-phase fault at bus-II**
Next, a 3-phase fault was induced at at bus-II of the IEEE 118 bus system at $t = 1s$, and cleared at $t = 2.5s$, with a clearing time of $1500ms$. Figure 5.6 shows the response of the large angular velocities of the faulty and the neighboring buses. The high-fidelity solution is shown by solid lines and the reduced solution by dashed lines; the absolute error is also shown as an inset in the same figure. As confirmed by the small values of absolute errors from Table 5.2, the two-time histories are indistinguishable from each other. As expected, the CPU times of reduced model obtained via NLMM-DMD method is much lower than POD-DEIM, as discussed earlier.

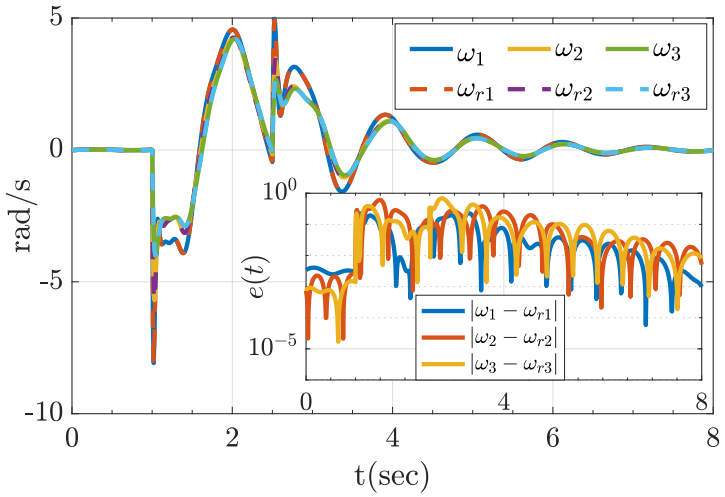

Figure 5.6: Second Scenario: Angular velocities of the buses I-III of the IEEE 118 bus system. Solid lines represent the full-models and dashed-lines represent the reduced-models. Inset: absolute errors between the original and reduced models

(iii) **Case III: 3-phase fault followed with line tripping**
Finally, a 3-phase fault was simulated at lines I and II of the IEEE-118 bus system, occurring at $t = 1s$. Then, at $t = 2.5s$ (after $1500ms$), the fault was manually cleared by isolating the faulty line from the rest of the system. This depicts the most typical fault isolation method in power systems. Figure 5.7 demonstrates this scenario with the same color coding as the previous case. We notice that after the fault is isolated, the system is able to recover and this was very effectively captured by the reduced model.

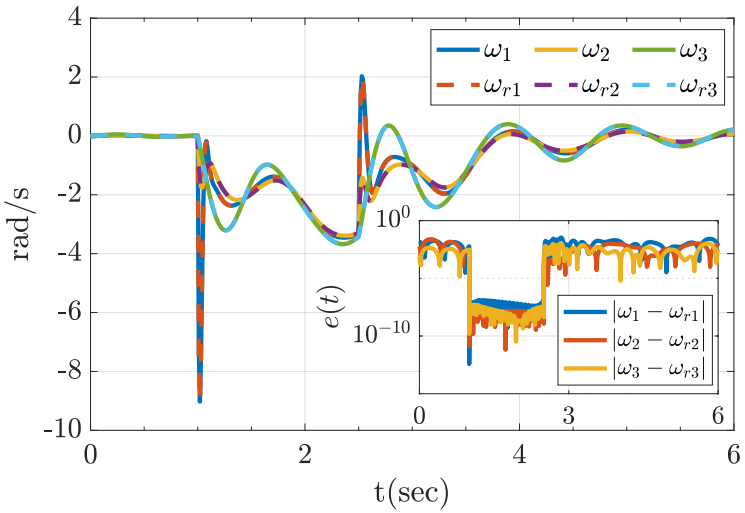

Figure 5.7: Third Scenario: Angular velocities of the buses I-III of the IEEE 118 bus system. Solid lines represent the full-models and dashed-lines represent the reduced-models. Inset: absolute errors between the original and reduced models

Table 5.2: Comparison of the CPU times and relative errors for different fault scenarios in IEEE 118 bus system: $n = 236, r = 20, m_d = m_D = 30$

	method	offline time (s)	online time (s)	total CPU time (s)	$\mathcal{L}_2-$norm error
Fault Case-I	FOM	-	-	59.37	-
	POD-DEIM	59.37	15	74.37	$5.7e^{-4}$
	NLMM-DMD	11.20	4.27	15.47	$2.3e^{-4}$
Fault Case-II	FOM	-	-	52.02	-
	POD-DEIM	52.02	8.51	60.53	$5.7e^{-3}$
	NLMM-DMD	16.62	3.20	19.82	$1.3e^{-3}$
Fault Case-III	FOM	-	-	69.96	-
	POD-DEIM	69.9	13.01	82.91	$5.6e^{-4}$
	NLMM-DMD	6.15	4.51	10.65	$7.7e^{-4}$

Next, we will consider the direct reduction of second-order model (5.13) using SONLMM-DEIM. This will preserve the second-order structure of the original model resulting in a second-order ROM of the form (5.15).

5.5.2 Ring-Grid model

The numerical setup considered here represents a "Ring-Grid" network having n-generators connected to an infinite bus (cf. Fig. 5.8). The governing dynamics are represented by (5.29). The reference node (slack node) is connected to the generators and is modeled as an infinite bus. It is assumed that both $|\mathbf{E}|$ and $\boldsymbol{\delta}$ are known apriori at the reference node. Furthermore, the following assumptions are made for simplicity (as given in Ref. [196]):

1. The network is loss-less.
2. The length of the transmission line connecting all the generators to the infinite bus is much larger than the lengths of lines between the individual generators. This ensures the least interaction between the generators and the infinite bus.
3. The length of transmission lines between the infinite bus and all the generators is the same.
4. The length of transmission lines between the generators is also the same.

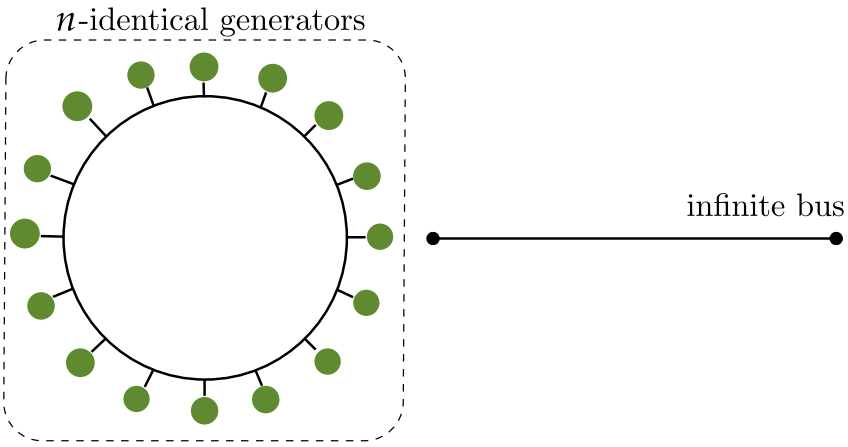

Figure 5.8: Diagrammatic visualization of a Ring-Grid network with n -generators (represented by green circles) connected to an infinite bus

Using the assumptions as above, the nonlinear term in (5.29) is modified as follows:

$$\mathbf{f}_i(\boldsymbol{\delta}) = b_1 \sin(\boldsymbol{\delta}_i) + b_2[\sin(\boldsymbol{\delta}_i - \boldsymbol{\delta}_{i+1}) + \sin(\boldsymbol{\delta}_i - \boldsymbol{\delta}_{i-1})] \quad i = 1, 2, ..., n \quad (5.34)$$

where b_1 represents the line susceptance between the generators and slack node, and b_2 is the line susceptance between two connected generators. The various model parameters used in the study are taken from [196] and are enlisted in Table 5.3.

Table 5.3: Ring Grid model: Parameters

Symbol	Description	Value
h_i	Mass of each alternator	1pu
d_i	Damping of each alternator	0.25pu
F_i	Power generated by each alternator	0.95pu
b_1	Susceptance between alternator and reference node	1pu
b_2	Susceptance between connected alternators	100pu
n	Number of alternators	1000

Three different test cases were simulated by solving the FOM (5.29), and the corresponding reduced models were obtained using SONLMM-DEIM via Algorithm 9. The FOMs and ROMs were integrated using a fully implicit second-order Euler's scheme [2] with a step size of $\delta t = 0.01s$. For comparison, POD ROM was used as reference. The numerical results are presented in Fig. 5.9 and Table 5.4. The three test cases are defined as follows:

(i) **Case-I: All nodes starting from an under perturbation condition**
For this scenario, all the nodes started from a non-equilibrium value $\boldsymbol{\delta}_i = 0.8$. The reduced basis via SONLMM was obtained using Algorithm 9 with $i = 1, k = 50$. The training input was generated by the exogenous system /signal generator $\boldsymbol{\zeta}(t) = 10\sin(\pi t + 50)$. For the application of DEIM, Algorithm 3 was used with $m_D = 10$. It is observed that the SONLMM-DEIM ROM was equally effective in approximating the original response like the POD with less computational times.

(ii) **Case-II: All nodes starting from an over perturbation condition**
Next, we simulated the case where all the nodes start from a non-equilibrium value of $\boldsymbol{\delta} = 1.12$, which corresponds to an over-perturbed scenario. It is seen that POD requires a basis time more than that of FOM due to the calculation of expensive snapshots matrix followed by SVD of a big matrix.

[2]The MATLAB implementation is available at `https://zenodo.org/record/5055753#.YN7-sOgzabg`

On the contrary, SONLMM-DEIM resulted in ≈ 96% reduction in offline time and around 50% reduction in online time.

(iii) **Case-III: All nodes starting from equilibrium condition**
Finally, we considered the case where all the nodes start from the equilibrium point of $\boldsymbol{\delta}_i = 1$. All the parameters regarding SO-NLMM and DEIM are kept the same as the previous cases. Again, SONLMM-DEIM performed better than POD in terms of computational time, however POD produced a more accurate approximation of the FOM.

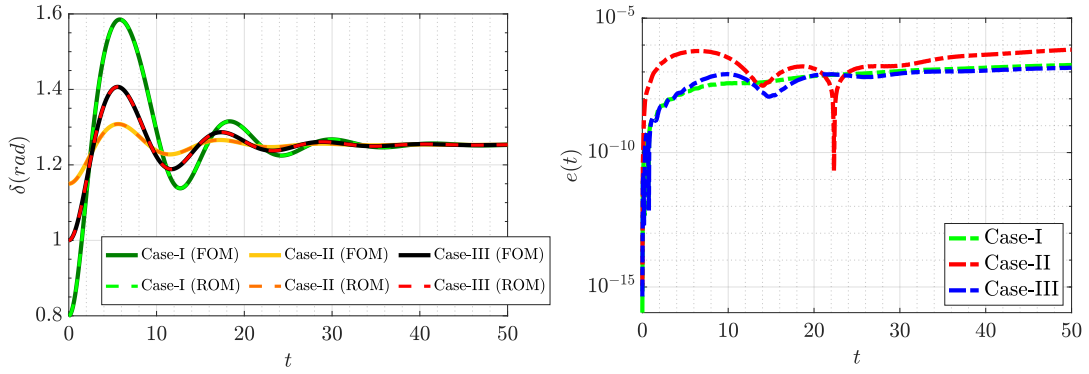

Figure 5.9: Left: Average of $\boldsymbol{\delta}$ of the FOM and SONLMM-DEIM ROM for three test case scenarios, Right: error profile of the ROMs for the three cases

Table 5.4: Ring Grid model: Comparison of ROMs for the three test cases

	method	size	CPU Time (s)			error in $\boldsymbol{\delta}$	
			basis	online	total	abs.	rel.
Case-I	Full Model	1000	-	-	103.43	-	-
	POD	15	109.98	33.37	143.36	$4.32e^{-7}$	$4.90e^{-9}$
	SONLMM-DEIM	15	5.87	20.03	25.91	$1.92e^{-5}$	$1.43e^{-7}$
Case-II	Full Model	1000	-	-	98.19	-	-
	POD	15	104.10	30.61	134.71	$5.65e^{-7}$	$3.92e^{-9}$
	SONLMM-DEIM	15	3.92	15.23	19.15	$8.01e^{-5}$	$6.75e^{-7}$
Case-III	Full Model	1000	-	-	98.54	-	-
	POD	15	104.16	32.01	136.18	$3.51e^{-5}$	$3.72e^{-9}$
	SO-NLMM-DEIM	15	3.69	15.41	19.11	$2.30e^{-5}$	$1.80e^{-7}$

5.6 Chapter Summary

In this chapter, we focused on the reduction of large-scale models in second-order form. First, we discussed the reduction of linear second-order models via moment-matching and the associated methods. Then, we described the notion of second-order moments for nonlinear systems. We derived the second-order Sylvester equation and employed simplifications from [295] for approximated moment-matching. Then, we discussed how DEIM could be used with SONLMM to obtain an efficient framework for second-order models. Finally, we validated the theory on certain power system models.

It can be concluded that SONLMM-DEIM is an efficient framework than POD. However, the performance of SONLMM-DEIM depends on the choice of signal generator employed. Thus, the signal generator chosen should excite the important dynamics of the system under study or constitute the representative *eigen functions* of the nonlinear system.

In the future, a more robust SONLMM can be derived where the choice of the signal generator becomes less heuristic and more automatic. Furthermore, a data-driven method such as DMD can be used instead of DEIM to reduce online time in a non-intrusive setup.

Part III

Nonlinear Parametric State-Space Systems

Chapter 6

Parametric Nonlinear Model Order Reduction

So far, in this thesis, we have discussed the reduction of large-scale nonlinear systems via moment-matching without any parametric dependencies. However, dynamical systems expressed as parameterized partial differential equations are ubiquitous in all branches of engineering and applied sciences. While standard MOR techniques are not robust to parametric variations, the process is exacerbated for large-scale problems with implicit parameter dependence. This part of the thesis will focus on large-scale models with parametric dependencies. Starting with this chapter, we lay the foundations of parametric model order reduction (PMOR) techniques and discuss a novel framework to obtain parametric reduced-order models (PROMs) that aim to preserve these parametric dependencies in the reduced models. We will demonstrate the theory on some nonlinear benchmark models, and towards the end of the chapter, we will provide some discussions and future perspectives.

6.1 Overview of parametric MOR

Parametric MOR aims to characterize system response for a broad class of problems for which the equations representing the system dynamics depend on a set of parameters. These parameters may enter the models in various ways, such as system geometry, material properties, system configuration, varying boundary, and initial conditions. This parametric dependence presents a unique set of challenges for MOR. Since a reduced model is usually obtained by solving the expensive full model, thus, one cannot afford to construct a new reduced model every time a parameter is changed. This requires repeated simulations or measurements of the full model for every parameter value, making untenable demands on computational resources and unfeasible design process. In addition, parametric dependencies don't always remain symbolically present in large-scale models. For example, in the finite element (FE) method, the geometric features are often

lost in the preprocessing meshing stage. As such, a new FE model is needed for every new parametric variation. Hence, the desired approach is to develop parametric MOR techniques to construct parametric reduced models that preserve the parametric dependency as symbolic variables and, at the same time, require lesser computational demands. The aim of PMOR is therefore two-fold;

1. To construct parametric ROM that approximates the original full-order dynamical system with high-fidelity over a range of parameters.

2. The parametric ROM should be computationally tractable to handle.

We begin by considering a class of large-scale nonlinear parametric systems represented in state-space form as:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathcal{N}(\mathbf{x}(t), \mathbf{u}(t) : \boldsymbol{\mu}), \\ \mathbf{y}(t) &= \mathcal{G}(\mathbf{x}(t)). \end{aligned} \tag{6.1}$$

Here $t \in [0, T]$ denotes the time with final time $T \in \mathbb{R}_+$, $\mathbf{x}(t) \in \mathbb{R}^n$ represent time-dependent parameterized state as the solution of (6.1), n being the dimension of the state-space model, $\mathbf{u} \in \mathbb{R}^m$ is the input (excitation) vector, $\mathbf{y}(t) \in \mathbb{R}^p$ is the measurement (observation) vector, $\boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^q$ is the input parameter vector, representative of physical variables characterizing an operating point, where the parameter space $\mathcal{P}$ represents a closed and bounded subset of the Euclidean space $\mathbb{R}^q, q \geq 1$, $\mathcal{N}() : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^q \to \mathbb{R}^n$ and $\mathcal{G} : \mathbb{R}^n \to \mathbb{R}^p$ represent the two nonlinear parametric mappings.

Parametric MOR seeks to replace such high-fidelity models by reduced models that feature a dramatically lower computational complexity yet retaining the most essential features of the map $\mathcal{N}$ and guaranteeing that the error between the solution of the reduced model and the original one stays below the desired value. The reduced model is usually obtained by applying a (Petrov) Galerkin projection method (cf. Section 2.1.1) that enforces the dynamics to evolve on a low-dimension subspace ($r << n$) of original state space, with:

$$\begin{aligned} \dot{\mathbf{x}}_\mathbf{r}(t) &= \mathbf{W}^T \mathcal{N}_r(\mathbf{V}\mathbf{x}_\mathbf{r}(t), \mathbf{u}(t) : \boldsymbol{\mu}), \\ \mathbf{y}_r(t) &= \mathcal{G}_r(\mathbf{V}\mathbf{x}_\mathbf{r}(t)), \end{aligned} \tag{6.2}$$

where $\mathbf{x_r} = \mathbf{V^T x} \in \mathbb{R}^r$ is the reduced state vector and $\mathbf{V}, \mathbf{W} \in \mathbb{R}^{n \times r}$ are the orthogonal projection matrices. It should be noted here that the idea behind PMOR is to allow any parameter variation in reduced subspace itself without having to repeat the reduction step i.e, the ROM derived should permit any parametric changes and at the same time should have a low computational requirement than the full order model.

Parametric MOR has seen a prominent success in recent years. For example, parametric reduced models are used to effectively capture the coupled structural and fluid dynamic behavior of aeroelastic performance of an aircraft over a range of flight conditions [8, 186, 231]. This has enabled rapid characterization of aircraft's flight conditions, calculations that would instead require weeks of computational time. Reduced parametric models have also been involved in the design and synthesis of semiconductor devices, see, e.g., Lee et al. [182], and Feng et al. [113, 111]. In control design, it is often desired to avoid designing a new controller every time a system parameter is changed, which otherwise can result in an unmanageable online controller. As such, significant contributions have emerged in this direction [233, 153, 7]. Furthermore, design and control also involve optimizations, in which the optimal configuration concerning a particular performance objective (e.g., minimal energy, maximum throughput, etc.) is needed. This requires repeated runs for varying parameter configurations. Thus, parametric ROMs play a crucial role in optimizations [34, 105, 312]. A combination of model reduction and domain decomposition has been developed for shape optimizations optimal control [12, 13]. Uncertainty qualification is another domain in which repeated model evaluations are required, e.g., in Monte Carlo sampling [65, 97], or PDEs with random coefficients [101].

Some of the famous PMOR methods include rational interpolation methods such as tangential interpolation method [47], moment-matching methods for parametric systems [51, 113, 160], balanced truncation methods [39], proper orthogonal decomposition (POD) based methods [232, 193], reduced basis (RB) method [251], and parametric MOR by matrix interpolation [224] for linear systems. The different strategies employed in these methods are discussed next.

6.2 Different strategies for generating parametric ROMs

In the following, we provide a brief survey of different strategies for constructing parameterized reduced models. Many of these methods are broadly applicable with any reduction approaches discussed in previous chapters. We consider $\boldsymbol{\mu}$ as the parameter of interest which belongs to a single domain $\mathcal{P} \subset \mathbb{R}^q$. However, various studies have adopted the strategy to first split $\mathcal{P}$ into multiple subdomains and then construct ROMs in each subdomain [10, 95, 99, 147, 228, 303].

6.2.1 Global basis approach

This class of methods construct a single pair of basis matrices $\mathbf{V}$ and $\mathbf{W}$ by sampling information over a range of parameters to produce near interpolants across the needed range of parameter values. The ROM takes the form as in (6.2). One of the common practices to obtain the global basis $\mathbf{V}$ and $\mathbf{W}$ is to concatenate the local basis matrices obtained for several parameter samples $\boldsymbol{\mu}_i : i = 1, 2, ..., q$.

Let $\mathbf{V}_1, ..., \mathbf{V}_q$ and $\mathbf{W}_1, ..., \mathbf{W}_q$ be the local basis matrices obtained at $\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_q$, then global version of $\mathbf{V}$ and $\mathbf{W}$ are constructed as:

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1, & \mathbf{V}_2, \cdots & \mathbf{V}_q \end{bmatrix} \text{ and } \mathbf{W} = \begin{bmatrix} \mathbf{W}_1, & \mathbf{W}_2, \cdots & \mathbf{W}_q \end{bmatrix}. \tag{6.3}$$

Note that, it is very likely that the local matrices may contain similar components, resulting in possible rank-deficient global basis matrices $\mathbf{V}$ and $\mathbf{W}$. To circumvent this issue, the concatenation step is frequently followed by an SVD or a rank revealing QR factorization. This eliminates the rank-deficient components from $\mathbf{V}$ and $\mathbf{W}$, resulting in global basis matrices with orthonormal columns. It is also worth noting that, while theoretically, it does not matter whether the local matrices $\mathbf{V}_i$ and $\mathbf{W}_i : i = 1, 2, ..., q$ are orthogonalized before the concatenation step as the reduced model is calculated by the range and not by a specific basis.

Any of the approaches discussed in Chapters 3 and 4 can be used to obtain the local basis matrices. The technique of choice, however, leads to distinct attributes in the reduced model. For instance, in linear parametric systems, the concatenation method is beneficial when the local basis matrices are constructed using the rational interpolation methods described in Section 2.2.6. According to Theorem 2.8, yet after concatenation, if the SVD performed on the concatenation matrices removes only the zero singular values, the resulting reduced model interpolates the original model at every parameter interpolation point and frequency combination used while constructing the local bases. This is in contrast to, say, if the local basis is obtained via the balanced truncation method. Therein, the result of the concatenation of basis no longer guarantees the reduced model to be balanced at parameter value $\boldsymbol{\mu}_i$. Nevertheless, because the reduced subspace retains the essential basis truncation information of the individual parameters, the concatenation of local balanced basis may still provide a decent reduced parametric model.

6.2.2 Local bases approach

Another possible and widely adopted approach is to construct several local basis matrices by sampling the parametric space at points $\boldsymbol{\mu}_i : i = 1, ..., q$. After collecting the local basis matrices $\mathbf{V}_i$ and $\mathbf{W}_i$ at these samples, one can proceed by either interpolating the local bases, the local reduced models, or local transfer functions (in case of linear systems). However, care has to be taken during interpolation. For instance, while interpolating local bases, a straightforward interpolation of entries in basis vectors might result in an interpolated quantity that doesn't retain desirable attributes. As pointed out in Ref. [8], the quantity to be interpolated should be the underlying subspace rather than the local basis. As a result, an improved technique interpolates the subspace belonging to $\mathbf{V}_i$ and $\mathbf{W}_i$ on a tangent space to a manifold of these subspaces tailored to preserve desired properties [8]. However, this method has a disadvantage that a new basis matrix for a new parameter $\boldsymbol{\mu}^*$ requires the evaluation of terms like $\mathbf{W}^T\mathbf{E}(\boldsymbol{\mu}^*)\mathbf{V}$ and $\mathbf{W}^T\mathbf{A}(\boldsymbol{\mu}^*)\mathbf{V}$ which depend on original system dimension n, making the operations expensive. This issue has

been addressed for affine parametric dependence in Ref. [273] wherein the terms that do not depend upon the parameters are precomputed beforehand. However, a feasible approach to treat the general nonaffine case is to interpolate reduced state-space components rather than the basis matrices themselves. This approach is adopted in Refs. [6, 9, 91, 189, 224]. In the following, we discuss the application of parametric MOR via matrix interpolation in nonlinear systems.

6.3 Nonlinear parametric MOR using matrix interpolatory framework

Parametric MOR via matrix interpolation for linear systems was first described by Panzer et al. [224, 129], and recently for nonlinear systems in Ref. [240]. The overall scheme consists of two main stages: the offline stage and the online stage. The offline stage is usually the most expensive one and requires a high computation requirement. The online stage, on the other side, is desired to be cost-effective to meet the requirements for real-time applications. During the offline stage, the parametric PDE is discretized in spatial domain using the usual finite difference (FD), finite elements or finite volume (FV) scheme. The discretization is carried out at different parametric instances $\boldsymbol{\mu}_i : i = 1, 2, ..., q$ to obtain a large-scale family of non-parametric models given as:

$$\dot{\mathbf{x}}_i(t) = \mathcal{N}_i(\mathbf{x}_i(t), \mathbf{u}(t)), \tag{6.4a}$$
$$\mathbf{y}_i(t) = \mathcal{G}_i(\mathbf{x}_i(t)). \tag{6.4b}$$

These large-scale models are then reduced by any nonlinear reduction scheme like snapshots based methods including proper orthogonal decomposition (POD) [306], dynamic mode decomposition (DMD) [241], singular value decomposition (SVD) [279]. Methods based on variational analysis [47, 112], or linearizations such as trajectory piece-wise linear method (TPWL) [249], bilinearization method [33, 231], quadratic method [81], or interpolatory/moment-matching methods, proposed in Refs. [237, 242, 239, 239, 25, 24] can also be used.

After constructing the local projection matrices $\mathbf{V}_i$ and $\mathbf{W}_i$ corresponding to each full-model, a family of reduced models can be then obtained as follows:

$$\dot{\mathbf{x}}_{r,i}(t) = \mathbf{W}_i^T \mathcal{N}_i(\mathbf{V}_i \mathbf{x}_{r,i}(t), \mathbf{u}(t)), \tag{6.5a}$$
$$\mathbf{y}_{r,i}(t) = \mathcal{G}_i(\mathbf{V}_i \mathbf{x}_{r,i}(t)), \tag{6.5b}$$

where $\mathbf{x}_{r,i} \in \mathbb{R}^r, \mathbf{V}_i, \mathbf{W}_i \in \mathbb{R}^{n \times r}, i = 1, 2, ..., q$.

Furthermore, the underlying nonlinearity in these models can be efficiently evaluated using hyperreduction schemes such as empirical interpolation method (EIM) [36], DEIM [80], best point method [217], the missing point estimation method [28] or the DMD method (as described in Chapter 4). Using the DEIM

via Algorithm 3, the resulting model takes the form:

$$\dot{\mathbf{x}}_{r,i}(t) = \boldsymbol{\beta}_{i,\text{DEIM}}\, \mathcal{N}_i(\boldsymbol{\gamma}_{i,\text{DEIM}}\mathbf{x}_{r,i}(t), \mathbf{u}(t)), \tag{6.6a}$$

$$\mathbf{y}_{r,i}(t) = \mathcal{G}_i(\mathbf{V}_i\mathbf{x}_{r,i}(t)), \tag{6.6b}$$

where the terms $\boldsymbol{\beta}_{i,\text{DEIM}} = \mathbf{W}_i^T\mathbf{U}_{\mathbf{d}i}(\mathbf{P}_i^T\mathbf{U}_{\mathbf{d}i})^{-1} \in \mathbb{R}^{r\times m_d}$ and $\boldsymbol{\gamma}_{i,\text{DEIM}} = \mathbf{P}_i^T\mathbf{V}_i \in \mathbb{R}^{m_d\times r}$ are both independent of n and hence can be precomputed. Note that each projection basis $\mathbf{V}_i, \mathbf{W}_i, \mathbf{U}_{\mathbf{d}i}, \mathbf{P}_i$ is obtained independently for each parametric variation $\boldsymbol{\mu}_i : i = 1, 2, ..., q$. Now, in order to obtain the reduced model for any new parametric choice outside the training set, we proceed with the interpolation of the local neighboring models.

A natural choice would be to take the weighted sum of the nearest neighbors. This choice of interpolation is equivalent to summing up their underlying systems of ODEs, and can be inaccurate if the respective state variables represent different physical quantities. Thus, for a meaningful interpolation of system matrices, the states of the reduced system have to be described in a generalized coordinate system by relating basis $\mathbf{V}_i$ to a reference basis. This is achieved by the state transformations $\mathbf{x}_{r,i} = \mathbf{T}_i^{-1}\mathbf{x}^*_{r,i}$ followed by pre-multiplying the matrix $\mathbf{M}_i \in \mathbb{R}^{r\times r}$ from the left which leaves the input-output behavior unchanged, i.e.,

$$\mathbf{x}^*_{r,i} = \underbrace{\mathbf{M}_i\boldsymbol{\beta}_{i,\text{DEIM}}}_{\boldsymbol{\Psi}^*_{r,i}}\mathcal{N}_i(\underbrace{\boldsymbol{\gamma}_{i,\text{DEIM}}\mathbf{T}_i^{-1}}_{\boldsymbol{\Omega}^*_{r,i}}\mathbf{x}^*_{r,i}(t), \mathbf{u}(t)), \tag{6.7a}$$

$$\mathbf{y}_i(t) = \mathcal{G}_i(\underbrace{\mathbf{V}_i\mathbf{T}_i^{-1}}_{\boldsymbol{\eta}^*_{r,i}}\mathbf{x}^*_{r,i}(t)), \tag{6.7b}$$

where matrices $\mathbf{M}_i = (\mathbf{V}_i^T\mathbf{R})^{-1}$, $\mathbf{T}_i^{-1} = \mathbf{R}^T\mathbf{V}_i$ and the matrix $\mathbf{R} \in \mathbb{R}^{n\times r}$ represents the universal subspace which contains r dominant directions of evolution. The columns of matrix $\mathbf{R}$ are constructed by first stacking all the local projection matrices in a single matrix $\mathbf{V}_{\text{cat}} = [\mathbf{V}_1,\ \mathbf{V}_2,\ ...\ \mathbf{V}_q] \in \mathbb{R}^{n\times(q\times r)}$ and then taking its SVD to retain r most significant directions.

During the online mode, the interpolated reduced model, for any new test parameter $\boldsymbol{\mu}^*$ selected outside the training set, is obtained by taking the weighted sum of the adjusted neighboring reduced-models as follows:

$$\dot{\mathbf{x}}^*_r(t) = \boldsymbol{\Psi}^*_r(\boldsymbol{\mu}^*)\mathcal{N}_r(\boldsymbol{\Omega}^*_r(\boldsymbol{\mu}^*)\mathbf{x}^*_r(t), \boldsymbol{\mu}(t)) \tag{6.8a}$$

$$\mathbf{y}_r(t) = \mathcal{G}_r(\boldsymbol{\eta}^*_r(\boldsymbol{\mu}^*)\mathbf{x}^*_r(t)) \tag{6.8b}$$

where

$$\boldsymbol{\Psi}^*_r(\boldsymbol{\mu}^*) = \sum_{i=1}^{q}\omega_i(\boldsymbol{\mu})\boldsymbol{\Psi}^*_{r,i},\ \ \boldsymbol{\Omega}^*_r(\boldsymbol{\mu}^*) = \sum_{i=1}^{q}\omega_i(\boldsymbol{\mu})\boldsymbol{\Omega}^*_{r,i},\ \ \boldsymbol{\eta}^*_r(\boldsymbol{\mu}^*) = \sum_{i=1}^{q}\omega_i(\boldsymbol{\mu})\boldsymbol{\eta}^*_{r,i} \tag{6.9}$$

and where

$$\sum_{i=1}^{q}\omega_i(\boldsymbol{\mu}) = 1, \quad \boldsymbol{\mu} \in \mathcal{P} \subset \mathbb{R}^q$$

The overall scheme is presented in Fig. 6.1

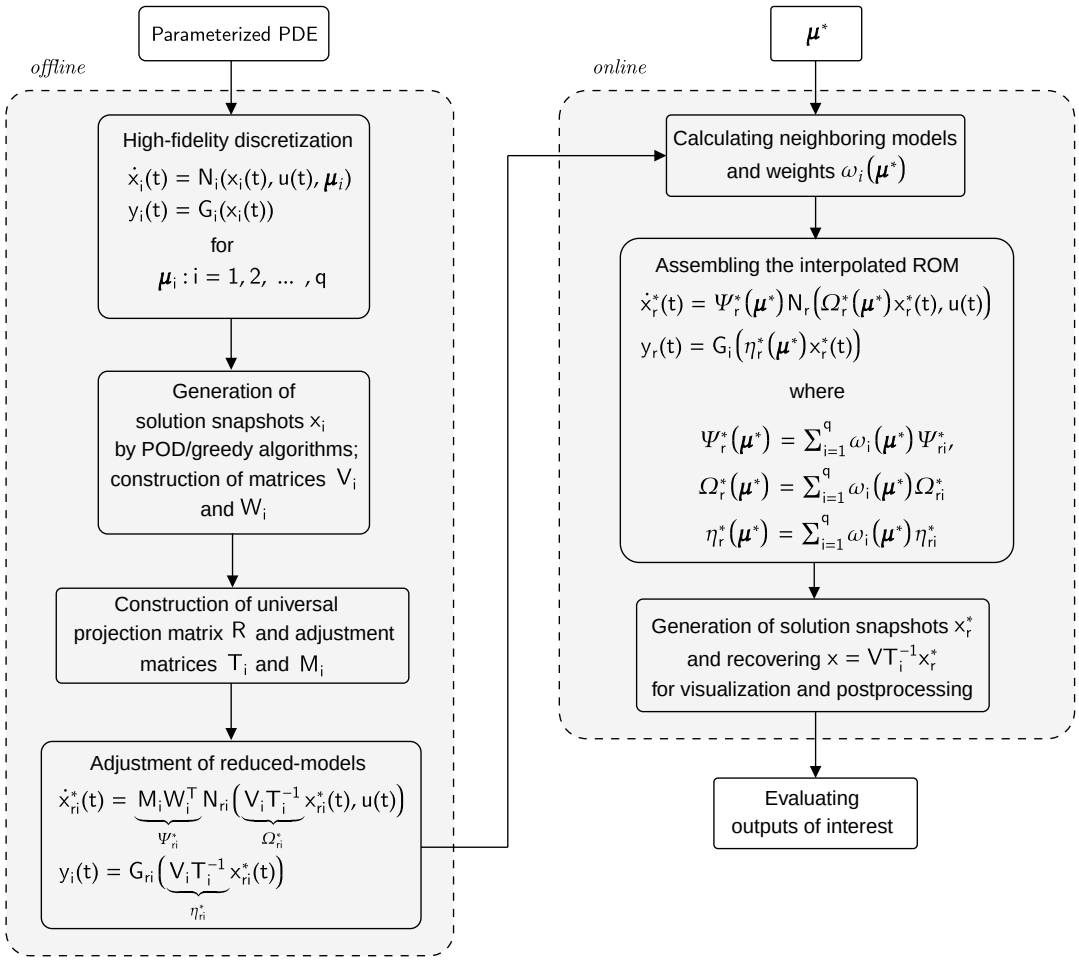

Figure 6.1: Nonlinear parametric MOR via matrix interpolation at a glance

6.4 Numerical validation

In the following, we will demonstrate the PMOR strategy on three benchmark parametric nonlinear systems (i) Nonlinear 1D Burgers' equation, (ii) Nonlinear 2D Burgers' equation, and (iii) 1D Reaction-Diffusion equation.

6.4.1 Nonlinear 1D Burgers' equation

To test the parametric scheme, we first considered the one-dimensional, parabolic, quasi-linear, viscous Burgers' equation:

$$\frac{\partial \mathbf{u}}{\partial t}(\mathbf{x}, t) + \mathbf{u}(\mathbf{x}, t)\frac{\partial \mathbf{u}}{\partial \mathbf{x}}(\mathbf{x}, t) = \frac{1}{Re}\left(\frac{\partial^2 \mathbf{u}}{\partial \mathbf{x}^2}(\mathbf{x}, t)\right) \quad \mathbf{x} \in (0, L), t \in [0, T], \quad (6.10)$$

subject to the following initial and boundary conditions:

$$\mathbf{u}(\mathbf{x},0) = 1 - \sin((2\pi/L)\mathbf{x}), \tag{6.11a}$$

$$\mathbf{u}(0,t) = \mathbf{r}(t);\ \frac{d\mathbf{u}}{d\mathbf{x}}(L,t) = 0. \tag{6.11b}$$

After a semi-discretization using finite difference scheme with n grid points, we obtained the following quadratic-bilinear structure of the state-space model:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{H}\ (\mathbf{x}(t) \otimes \mathbf{x}(t)) + (\mathbf{N}\mathbf{x}(t) + \mathbf{B})\mathbf{r}(t), \tag{6.12}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$ and $(\mathbf{x} \otimes \mathbf{x})$ is the *Kronecker* product. The matrices $\mathbf{A}, \mathbf{H}$, and $\mathbf{N}$ can be seen in Appendix (A.1). A single control input $\mathbf{r}(t)$ was applied on the left boundary and the desired output was measured on the right boundary. The various parameters used during the simulation are enlisted in Table 6.1

Table 6.1: 1D Burgers' equation: Parameters of the selected scenarios

Method	Parameters
Full model	$n = 500, L = 1, t \in [0, 3s], dt = 0.01s, \mu = [0, 1]$
ROM_d	NLMM: $r = 20, \dot{\boldsymbol{\zeta}}(t) = \tanh(t) + 5e^{-4}, i = 1, K = 101$ DEIM: $m_d = 10, n_s = 250$
Test input	$\mathbf{r}(t) = 0.5(\cos(2\pi/10)t)$

Table 6.2: 1D Burgers' equation: Weights and neighboring samples for the two test viscosity values

Test Sample Point	Neighboring Sample Points	weights chosen
μ_1 =0.857	0.78 & 0.89	$\omega_1 = 0.3, \omega_2 = 0.7$
μ_2 =0.186	0.12 & 0.23	$\omega_1 = 0.4, \omega_2 = 0.6$

Now, ten FD models, with size $n = 500$ were generated by varying the viscosity parameter ($\mu = 1/Re$) between $\mu = 0$ and $\mu = 1$, using an implicit *Euler's* scheme ($dt = 0.01$) which is accurate but expensive. The reduced models of size $r = 20$ corresponding to each of these models were obtained via NLMM (Algorithm 4). The training signal was provided by a single signal generator ($j = 1, K = 101$) with dynamics $\dot{\boldsymbol{\zeta}}(t) = \tanh(t) + 5e^{-4}$ and the nonlinearity was approximated via DEIM (Algorithm 3), with $m_D = 10$. To test the interpolation scheme, the model at $\mu = 0.857$ was obtained by interpolating the neighbouring reduced models at $\mu = 0.78$ and $\mu = 0.89$ (which took an online CPU time of $4.12s$ as compared

to 509s for the FOM). Another model for $\mu = 0.186$ was similarly obtained by interpolating the ROMs at $\mu = 0.12$ and $\mu = 0.23$ (online CPU time of 4.28s) with appropriately chosen weights as given in Table 6.2. The two test parameters selected, cover the lower and higher range of the parametric space of interest. The directly reduced models (ROM$_d$) corresponding to $\mu = 0.857$ and $\mu = 0.186$ were also stored for the sake of comparison. Figure 6.2 shows the comparison for the output response of 1D *Burgers'* equation for the two test parameter values and Fig. 6.3 depicts the relative $\mathcal{L}_2$ norm errors between the FOM and reduced models. It is seen clearly that the interpolated ROM accurately replicated the response of FOM just like the directly reduced model with relatively less error.

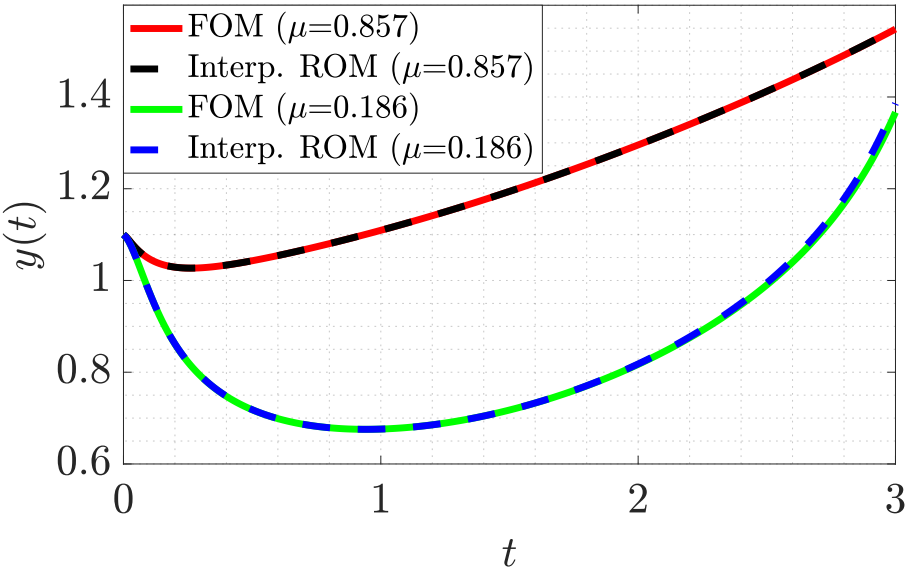

Figure 6.2: 1D Burgers' equation: Comparison of output responses for two test parameter values.

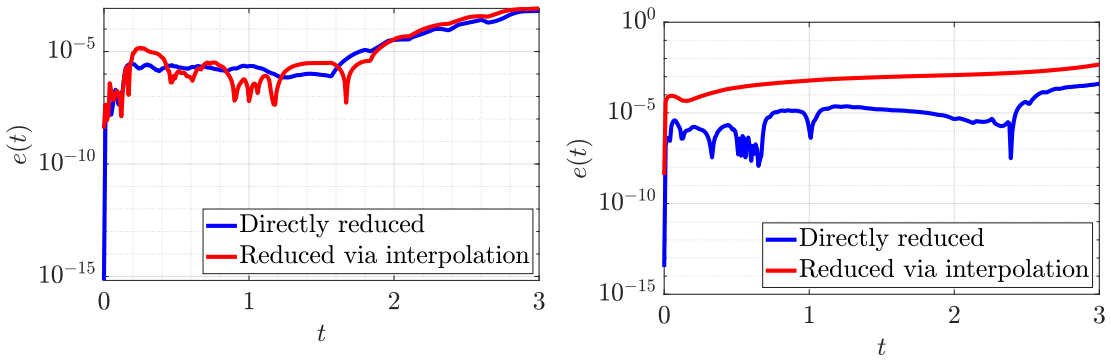

Figure 6.3: 1D Burgers' equation: Comparison of relative $\mathcal{L}_2$ norm error between FOM and ROMs for (left) $\mu = 0.857$, (right) $\mu = 0.186$

6.4.2 Nonlinear 2D Burgers' equation

Next, we considered the 2D version of the Burgers' equation. The governing equations are described by the following coupled PDEs:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\frac{\partial \mathbf{u}}{\partial \mathbf{x}} + \mathbf{v}\frac{\partial \mathbf{u}}{\partial \mathbf{y}} = \frac{1}{Re}\left(\frac{\partial^2 \mathbf{u}}{\partial \mathbf{x}^2} + \frac{\partial^2 \mathbf{u}}{\partial \mathbf{y}^2}\right), \tag{6.13a}$$

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{u}\frac{\partial \mathbf{v}}{\partial \mathbf{x}} + \mathbf{v}\frac{\partial \mathbf{v}}{\partial \mathbf{y}} = \frac{1}{Re}\left(\frac{\partial^2 \mathbf{v}}{\partial \mathbf{x}^2} + \frac{\partial^2 \mathbf{v}}{\partial \mathbf{y}^2}\right), \tag{6.13b}$$

$$(\mathbf{x}, \mathbf{y}) \in \Omega = (a, b) \times (c, d),\ t \in (0, T),$$

subject to the initial conditions:

$$\mathbf{u}(\mathbf{x}, \mathbf{y}, 0) = \phi_1(\mathbf{x}, \mathbf{y}),$$
$$\mathbf{v}(\mathbf{x}, \mathbf{y}, 0) = \phi_2(\mathbf{x}, \mathbf{y}),$$

and the boundary conditions:

$$\mathbf{u}(a, \mathbf{y}, t) = f_1(\mathbf{y}, t);\ u(b, \mathbf{y}, t) = f_2(\mathbf{y}, t),$$
$$\mathbf{u}(\mathbf{x}, c, t) = f_3(\mathbf{x}, t);\ u(\mathbf{x}, d, t) = f_4(\mathbf{x}, t),$$
$$\mathbf{v}(a, \mathbf{y}, t) = g_1(\mathbf{y}, t);\ v(b, \mathbf{y}, t) = g_2(\mathbf{y}, t),$$
$$\mathbf{v}(\mathbf{x}, c, t) = g_3(\mathbf{x}, t);\ v(\mathbf{x}, d, t) = g_4(\mathbf{x}, t),$$

where $\mathbf{u}(\mathbf{x}, \mathbf{y}, t)$ and $\mathbf{v}(\mathbf{x}, \mathbf{y}, t)$ are the two velocity components. The spatial domain is discretized uniformly into $n_x - 1$ and $n_y - 1$ intervals in $\mathbf{x}$ and $\mathbf{y}$ direction, respectively for a $n_x \times n_y$ grid making the total dimension of the problem $2(n_x - 1)(n_y - 1)$. The initial and boundary conditions are derived from the exact traveling wave solutions as given in Ref. [115].

$$\mathbf{u}(\mathbf{x}, \mathbf{y}, t) = \frac{3}{4} - \frac{1}{4[1 + \exp((-4\mathbf{x} + 4\mathbf{y} - t)Re/32)]}, \tag{6.14a}$$

$$\mathbf{v}(\mathbf{x}, \mathbf{y}, t) = \frac{3}{4} + \frac{1}{4[1 + \exp((-4\mathbf{x} + 4\mathbf{y} - t)Re/32)]}. \tag{6.14b}$$

The finite-difference scheme of 2D Burgers' equation can be seen in Appendix A.2. An input of $r(t) = 0.5\sin(t)$ was applied on the left boundary of u and v. Similar to the one-dimensional case, fifteen FD models were generated for a spatial grid size of 30×30, by varying Re between $Re = 0$ and $Re = 50$ using a fully implicit scheme Ref. [30], for an equally spaced interval of 0.03 between $[0, T], T = 5$. The corresponding reduced models were obtained using $r = 15$ NLMM basis constructed via Algorithm 4, for each full-order model and $m_d = 50$ DEIM basis were generated for approximating the nonlinearity using Algorithm 3. The signal generator with dynamics $\dot{\boldsymbol{\zeta}}(t) = -e^{-t} - 0.08$ was chosen to excite the system for $i = 1$ and $K = 20$ in this case.

During the online stage, the model at $Re = 24.64$ was obtained by interpolating the neighbouring reduced models at $Re = 21.42$ and $Re = 25$ with weights $\omega_1 = 0.1$ and $\omega_2 = 0.9$ respectively which took an online CPU time of $1.8s$ as compared to the $84.99s$ for the original full model. Figure 6.4 shows the solution of u of directly reduced ROM (ROM_d) and the interpolated ROM at time-step $nt = 50, 100$ alongwith the absolute of errors between the two. It is seen that the interpolated performs satisfactorily just like the directly reduced ROM for the test parameter outside the training set.

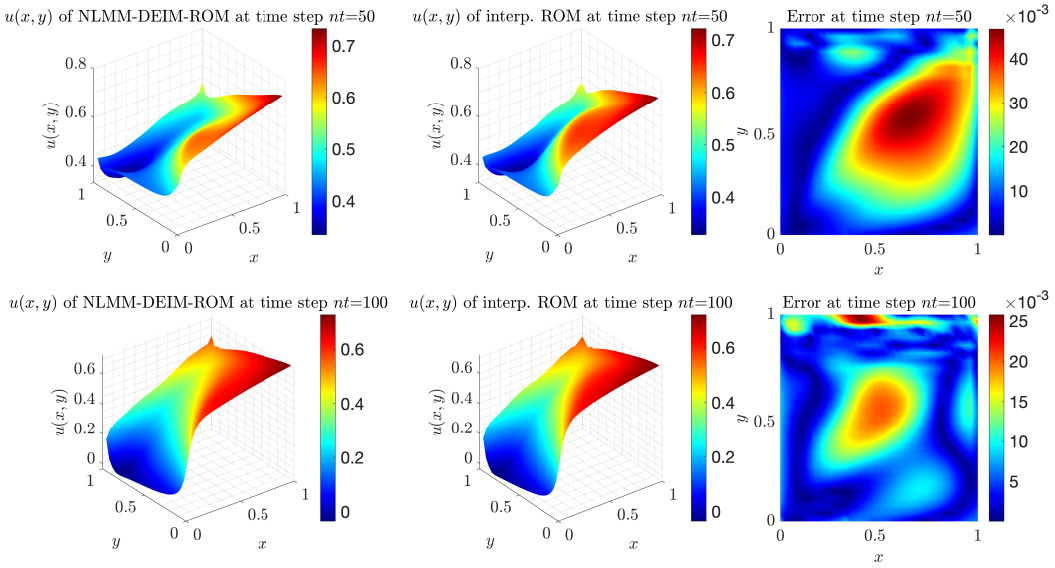

Figure 6.4: 2D Burgers' equation:Comparison of $\mathbf{u}$ between directly reduced model and the Interp. model at: (top) $nt = 50$, $(bottom)$ $nt = 100$ for $Re = 24.64$ in the spatial domain $\Omega = [0, 1] \times [0, 1]$.

Table 6.3: 2D Burgers' equation: Parameters of the selected scenarios

Method	Parameters
Full model	$n = 1682, (a, b) \times (c, d) = (0, 1) \times (0, 1), t \in [0, 5s], Re = [0, 50]$
ROM_d	NLMM: $r = 15, \dot{\boldsymbol{\zeta}}(t) = -e^{-t} - 0.08, i = 1, K = 20$
	DEIM: $m_d = 50, n_s = 250$
Test input	$\mathbf{r}(t) = 0.5\sin(t)$

6.4.3 1D Reaction-Diffusion equation

Finally, we considered the 1D reaction-diffusion equation. The governing PDE is given as:

$$\frac{\partial \vartheta}{\partial t}(\mathbf{x},t) = \theta \frac{\partial^2 \vartheta}{\partial \mathbf{x}^2}(\mathbf{x},t) - \lambda(\vartheta^3(\mathbf{x},t) - \vartheta(\mathbf{x},t)), \tag{6.15}$$

with initial condition and boundary conditions as:

$$\begin{aligned} &\vartheta(\mathbf{x},0) = 1 - \sin(\pi \mathbf{x}/L), && \mathbf{x} \in (0,L), \\ &\vartheta(0,t) = \mathbf{u}(t), \qquad \frac{\partial \vartheta}{\partial \mathbf{x}}(L,t) = 0, && t \in (0,T). \end{aligned}$$

Here θ and λ are the two parameters of interest, which adjust the relative balance between the diffusion term and the nonlinear term. An input of $\mathbf{u}(t) = 0.5(\cos(\pi t) + 1)$ was applied on the left boundary and the output $\mathbf{y}(t)$ was assumed to be the right boundary. The various parameter used for this case are enlisted in Table 6.4

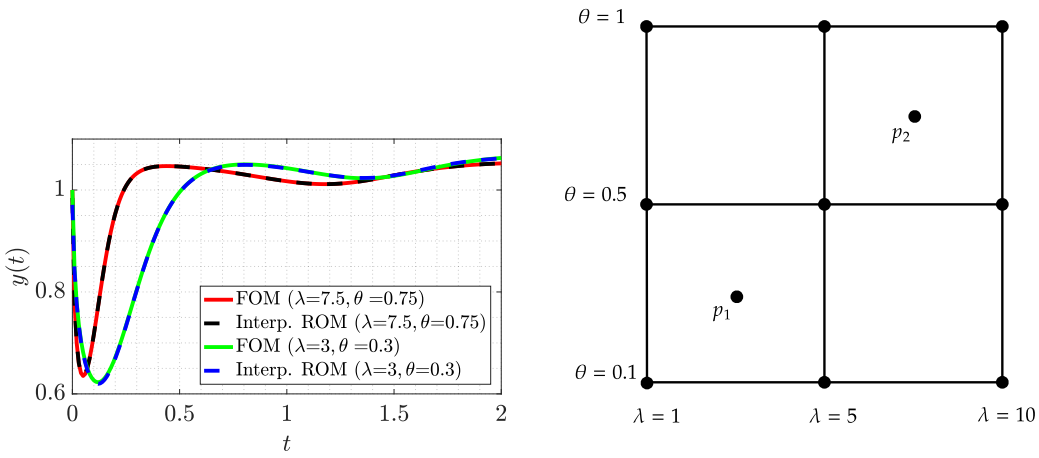

Figure 6.5: 1D Reaction Diffusion equation: (left:) Output response (right:) Illustration of 2D parametric space

Table 6.4: 1D Reaction-Diffusion equation: Parameters used

Method	Parameters
Full model	$n = 1000, L = 1, \lambda = 1, t \in [0, 2s], \theta = [0.1, 1], \lambda = [1, 10]$
NLMM	$r = 30, \dot{\boldsymbol{\zeta}}(t) = 0.5e^{-t/5} + \pi\cos(\pi t)e^{-t/5} + \sin(\pi t)e^{-t/5}, i = 1, K = 61,$ $t_k \in [0, 10s]$
DEIM	$m_d = 10, n_s = 200$
Test Input	$\mathbf{u}(t) = 0.5(\cos(\pi t) + 1)$

For the application of the parametric scheme, nine FD models of size $n = 1000$ were obtained using an implicit *Euler's* scheme with equally space interval of $dt = 0.01$, for $\theta = \{0.1, 0.5, 1\}$ and $\lambda = \{1, 5, 10\}$. The reduced models of size $r = 30$ were generated for each sample via NLMM-DEIM (cf. Algorithm 5). To test the proposed scheme, two test parameter values of $p_1 = (\lambda_{\text{test}}, \theta_{\text{test}}) = (3, 0.3)$ and $p_2 = (7.5, 0.75)$, were chosen which cover the lower and higher variations of parametric space. Under this setting, the interpolation was performed between four neighboring reduced models as depicted in Fig. 6.5. The weights and the neighboring ROMs used for interpolation are enlisted in the Table 6.5.

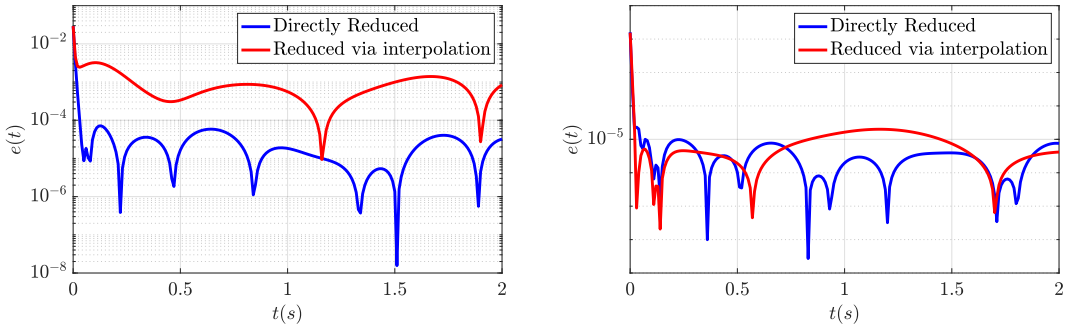

Figure 6.6: 1D Reaction-Diffusion equation: Comparison of relative $\mathcal{L}_2$ norm error between FOM and ROMs for: (left) $\lambda_{\text{test}} = 3, \theta_{\text{test}} = 0.3$, (right) $\lambda_{\text{test}} = 7.5, \theta_{\text{test}} = 0.75$

The online CPU times for both the two test parametric models were recorded around $3.56s$ which was way less than the full model requirement of $110s$. Figure 6.5 shows the output response (measured on the right boundary) of 1D Reaction-Diffusion model for the two test parameters and Fig. 6.6 presents the relative $\mathcal{L}_2$ norm errors between FOM and the reduced models. As can be seen in the two parameter setup also, the directly reduced ROM and the interpolated ROM match to a high degree of accuracy.

Table 6.5: 1D Reaction-Diffusion equation: Weights and neighboring samples for the two test samples values.

Test Sample Points	Neighbouring Samples	Weights Chosen
$\theta = 0.3,\ \lambda = 3$	$\theta = \{0.1, 0.5\};$ $\lambda = \{1, 5\}$	$\omega_1 = 0.1, \omega_2 = 0.3$ $\omega_2 = 0.4, \omega_4 = 0.2$
$\theta = 0.75, \lambda = 7.5$	$\theta = \{0.5, 1\};$ $\lambda = \{5, 10\}$	$\omega_1 = 0.1, \omega_2 = 0.3$ $\omega_2 = 0.4, \omega_4 = 0.2$

In order to test the efficacy of the proposed scheme across the entire parametric space, we also obtained the average errors for all test models, which is presented in Fig. 6.7. We can observe that the parametric ROMs obtained via the proposed scheme produce high-fidelity approximations of the actual responses across a wide range of parametric variations.

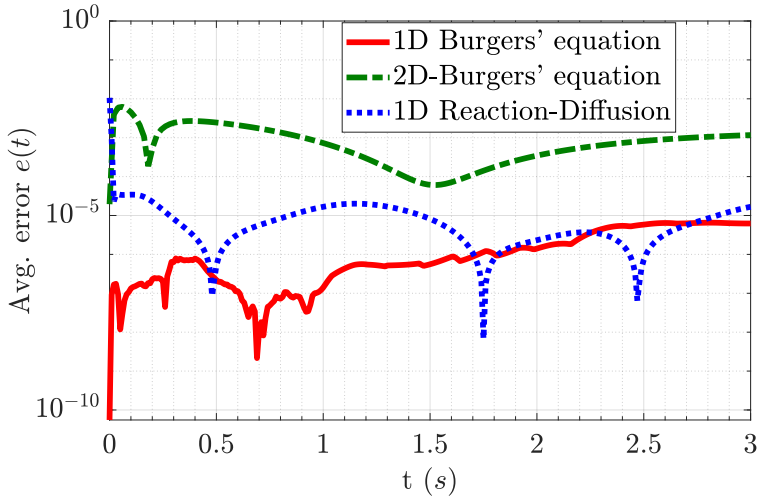

Figure 6.7: Average errors of the different parametric values across the parametric space.

6.5 Discussions and limitations

The parametric scheme described above benefits from generating a new FD model for every parametric variation. Once the local basis matrices and the global subspace are obtained (which is to be calculated only once), the interpolation of the neighboring nonlinear models is determined online. Thus, the proposed method meets the requirements of parametric MOR. Furthermore, the proposed interpolation scheme is independent of any nonlinear projection-based reduction technique. One can also employ a conventional POD-DEIM method to obtain the family of reduced models for various parameters. However, we advocate the use of the NLMM-DEIM framework because it yields notable savings in the offline calculation of the projection basis.

Kindly note that the dimension of the reduced models (r), obtained via NLMM, depends on the time-instances (K in Algorithm 4) chosen to capture the evolving dynamics of the selected signal generator. This further depends on how quickly the signal generator changes (e.g., exponential growth/fall). Once the values of K and r were obtained for a particular signal generator, then it needed to be tested across multiple parameters. This was quantified by repeated reductions along the parametric span of the problem. At the end, any redundant columns of the NLMM basis were deflated using the SVD scheme. Though we tried using different dimen-

sions of NLMM and DEIM basis during our tests, this only resulted in increased offline and online CPU times without much improvement in the approximating quality of the interpolating ROM, which mainly depends on taking more samples in sensitive zones. Therefore, we chose the lowest dimensions of the NLMM basis and DEIM, which provided satisfactory results without burdening the CPU. This is evident from Fig. 6.8.

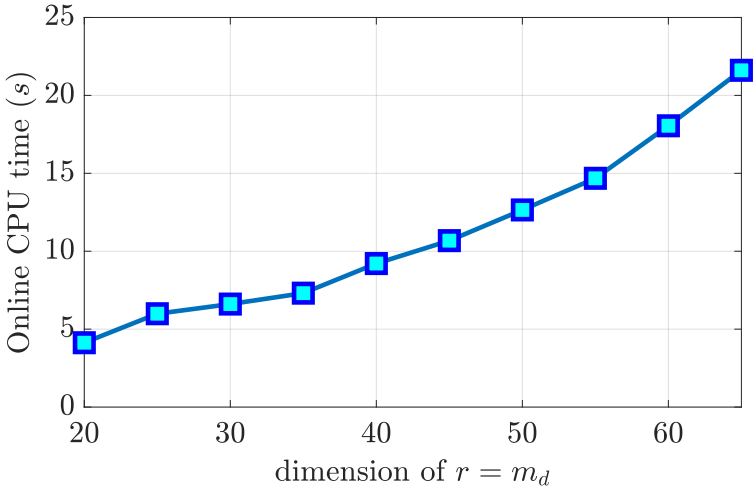

Figure 6.8: CPU time for different ranks of r, m_d for 1D Burgers' equation

As far as the identification of the nearest neighbors for a particular parameter is concerned, it was determined by their proximity to the desired parameter value. That means for a single-dimensional parametric space, the models on each side of the required parameter are the neighbors. For two-dimensional space, the models enclosing the desired parameter are the neighbors. The weights assigned to each neighboring model are inversely proportional to the distances of the selected parameter value from its respective neighbors.

Though the proposed scheme performs satisfactorily by taking a uniform sampling across the parametric space, it suffers from certain limitations. First, the samples are chosen manually without any information from the model. This can sometimes lead to oversampling or undersampling scenarios of the parametric space. Hence an adaptive sampling scheme with more samples taken at highly sensitive zones can be more helpful. This problem will be addressed in the next chapter. Furthermore, the proposed strategy has been demonstrated for models only for up to two parameters. The scheme can be extended for models with higher dimensional parametric space. However, care must be taken to avoid the exponential growth of basis due to the "curse of dimensionality".

6.6 Chapter Summary

In this chapter, we focused on reducing large-scale models with parametric dependencies. We highlighted some issues while simulating such models for optimization and control and how conventional MOR methods fail to produce reliable results for these models. We also discussed some common design strategies of parametric ROMs. Towards this aim, we proposed a PMOR framework for large-scale nonlinear dynamical systems based on matrix interpolation. The method is based on the offline-online framework. First, a family of large-scale models is generated for the parametric range of interest. The resulting models are then reduced via NLMM-DEIM in the offline stage, followed by back projecting on a common universal subspace. Finally, the nonlinear reduced model for any new parameter is obtained by interpolating the neighboring nonlinear models using the matrix interpolation scheme inspired by the matrix interpolation reported earlier. In the end, we presented some numerical simulations demonstrating the use of this scheme on certain nonlinear models. The results thus obtained are in agreement with the theoretical concepts. The next chapter will focus on the main issue among all PMOR methods, i.e., the adaptive parametric sampling.

Chapter 7

Adaptive Parametric Sampling Scheme for Nonlinear Systems

In the previous chapter, we examined the parametric MOR in nonlinear systems. We demonstrated the parametric MOR framework whereby PROMs are obtained via interpolation of neighboring models. However, the issue of parametric sampling remained elusive. In this chapter, we discuss different parametric space sampling strategies to construct the local basis. Furthermore, to enhance the approximating quality of the interpolated reduced models, we introduce a generalized framework for adequate sampling of the parametric space. Finally, we will substantiate the observations through numerical simulations of time-dependent parametric models originating from circuit theory and MEMS.

7.1 Sampling schemes for reduction of parametric systems

One of the most challenging step in the matrix interpolatory MOR method, discussed in previous chapter, is the sampling of parametric space. It involves selecting an adequate number of samples to cover the entire parametric region while maintaining the offline computational costs to a minimum. For problems involving a small number of parameters, a uniform or random sampling, grid-based sampling, Latin hypercube sampling, or a logarithmic sampling are the most simplistic approaches. For a moderate number of parameters, a sparse-grid sampling will be a practical approach. However, problems with a much higher-dimensional parametric space ($q > 10$) require more sophisticated sampling approaches such as problem-aware adaptive search of the parametric space. The drawback in a uniform sampling scheme is that if few samples are selected in the initial training set, the original solution manifold may not be adequately represented, leading to a poor ROM with significant errors. If the parametric space is too fine, the offline times can be prohibitively long, especially for higher parametric space dimensions. On the other hand, logarithmic sampling works better if the complex frequency

is selected as the running parameter and often results in under-sampling in most sensitive zones and oversampling in less important regions. Although the Latin hypercube sampling remains tractable for moderate values of parameters, it requires a large number of sample points to ensure sufficient coverage. As far as the random sampling approach is concerned, it results in a non-uniform parametric distribution with no relation to the nature of parametric dependence. Hence, these standard sampling techniques fall under the category of "blind" sampling or "open-loop gridding." As such, an adaptive sampling scheme in which the underlying system dynamics guide the selection of appropriate parameter samples is highly desirable. This allows to moderate the refinement procedure iteratively. Furthermore, it is desired that the sampling scheme should avoid the curse of dimensionality for higher-dimensional parameter spaces.

Towards this direction, the greedy sampling method was first introduced by Prud'Homme et al. [234] which was further developed in the RB framework for elliptical PDEs [300], parabolic PDEs [134], the steady incompressible Navier-Stokes equation [299], the rational interpolation methods [112], and with POD methods [200, 136, 52]. The greedy methods are based on the premise of adaptively selecting samples by discovering the locations in parametric space where the estimate of the error in reduced-model is maximum. Afterward, the full model is generated as these parametric locations for obtaining new information to update the reduced models accordingly. In a general setup, the actual reduced-model error is used as an indicator to find the worst-case parameter. This, however, leads to a computationally intractable algorithm since the full-order model is repeatedly solved at many parameter samples. Therefore the most adopted approach is using a posteriori error estimates for a given model [136]. For a low-dimensional parametric regime, the process of finding the parameter with the highest error indicator or error estimator is conducted using a simple grid search. For higher parameter dimensions, the greedy sampling scheme is reformulated as a sequence of adaptive model-constrained optimization problems [200]. These greedy sampling schemes, though performing satisfactorily for a range of models, suffer from a few drawbacks. Firstly these optimization problems have an explicit solution and become convex only for the special case of a linear map between parameters and outputs. Generally, the optimization problems are nonconvex and often converge to local maxima. Secondly, the error indicators are not always the true representations of the actual error. For further details, we refer the reader to Ref. [53] for an excellent review on this topic.

Apart from greedy search methods, the appropriate selection of samples in parametric space can be made by measuring the system's sensitivity to parametric variations. A fundamental assumption in matrix interpolatory PMOR is that the system dynamics remain restricted to the same dimensional subspace irrespective of the parametric variations. This assumption allows measuring the system's sensitivity to parametric variations by observing the change in attributes of the projection matrices $\mathbf{V}(\boldsymbol{\mu})$ to parameter changes. This can be quantified by mea-

suring the change in the distance between the successive projection subspaces. Since, for q uniformly placed parameter sample points $(\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_q)$, the corresponding projecting subspaces $(\mathbf{V}_1, \mathbf{V}_2, ... \mathbf{V}_q)$ maybe not be equally apart (cf. Fig. 7.1). As such, a parametric span with a larger subspace distance implies a highly sensitive zone; hence more samples (and ROMs) must be added. Similarly, less sensitive zones can be identified.

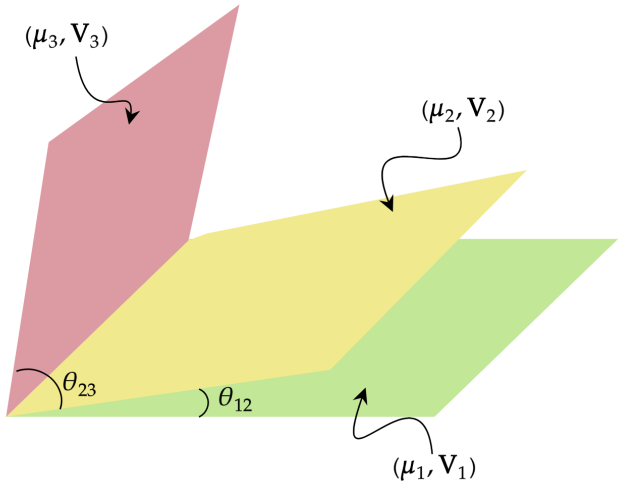

Figure 7.1: Illustration of a parametric subspaces

7.2 Adaptive sampling scheme

To begin with the adaptive sampling, we discuss the idea of *subspace angles* or *principal angles*.

7.2.1 Notion of distance between subspaces

The notion of distance or gap between subspace is well known in geometric algebra and is described using principal angles. To illustrate the idea, consider two r-dimensional subspaces $\mathbf{V}$ and $\mathbf{W}$ of an n-dimensional Euclidean vector spaces $\mathbb{R}^n$. These subspaces are spanned by a set of r linearly independent vectors given as:

$$\begin{aligned} \mathbf{V} &= \text{span}\{v_1, .., v_r\} \subset \mathbb{R}^n, \\ \mathbf{W} &= \text{span}\{w_1, .., w_r\} \subset \mathbb{R}^n. \end{aligned}$$

The conventional approach to obtain the subspace angle $\theta_{\mathbf{V},\mathbf{W}}$ between subspaces $\mathbf{V}$ and $\mathbf{W}$ is to first arrange the vectors as column vectors into two matrices as follows:

$$\mathbf{A} = \begin{bmatrix} v_1 & v_2 & ... & v_r \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} w_1 & w_2 & ... & w_r \end{bmatrix} \in \mathbb{R}^{n \times r}. \tag{7.1}$$

Then, standard matrix algebra methods like singular value decomposition and QR decomposition is applied to obtain:

- r pair of singular unit vector $\mathbf{v}_k, \mathbf{w}_k$, and
- r singular values $\sigma_k = \cos(\theta_k) = \mathbf{v}_k.\mathbf{w}_k$.

However, this is a computationally intensive approach. Another feasible method, as demonstrated by Wedin [304], is to find the r principal angles $\theta_i, (1 < i < r)$ that characterize the angular relationship between the subspaces $\mathbf{V}$ and $\mathbf{W}$ as follows:

$$\cos(\theta_{\mathbf{V},\mathbf{W}}) = \prod_{i=1}^{r} \cos(\theta_i)$$

where the inner product is canonically defined on the Grassmann algebra corresponding to the geometry in $\mathbb{R}^n$.

Alternatively, the principal angles can be recursively defined as:

$$\cos(\theta_{\mathbf{V},\mathbf{W}}) = \max_{a \in \mathbf{V}} \max_{b \in \mathbf{W}} a^T b,$$

subject to

$$\left\| a \right\| = \left\| b \right\| = 1, a^T a_i = b^T b_i = 0, \ i = 1 : r - 1,$$

where the vectors $a = \{a_1, ..., a_r\}$ and $b = \{b_1, ..., b_r\}$ are the principal vectors. The expression for largest subspace angle between V and W is then given as:

$$\theta_{\mathbf{V},\mathbf{W}} = \arccos(\sigma_r) = \arcsin(\sqrt{1 - \sigma_r^2}),$$

where σ_r is the smallest singular value of $\mathbf{V}^T\mathbf{W}$. Similarly the distance between other subspaces can be obtained.

In what follows, we use the distance between subspaces to adaptively detect the parametric space's sensitive regions to resample the grid in the offline stage. The user only needs to provide the maximum subspace tolerance value $\theta_{\max}$ to be used during resampling. As a result, the same FD model will be used until the tolerance angle is reached, and beyond that, a new model is obtained. The automatic refining scheme is described as follows:

7.2.2 Single parameter case

For systems having a single parameter, a uniformly distributed sample space is selected initially. Obviously, this sampling scheme shares the disadvantage mentioned before, however it is used to obtain a rough sample distribution that can be refined later. After obtaining the projection matrices $\boldsymbol{V}_i$ $(i = 1, .., q)$ corresponding to each parametric value $\boldsymbol{\mu}_i$, each subspace angle $\theta_{i,i+1}$ is recorded and compared with the maximum tolerance angle $\theta_{\max}$ and a ratio $l_{i,i+1} = \theta_{i,i+1}/\theta_{\max}$ is obtained [294]. Then, a ceiling function of this ratio provides an intuitive way of refining the initial parametric span, i.e., if this ratio equals one, no further division of parametric span is needed otherwise the parametric interval of $[\boldsymbol{\mu}_i, \boldsymbol{\mu}_{i+1}]$

is divided into $l_{i,i+1}$ intervals, and the process repeats again. The algorithm terminates when all the ratios $l_{i,i+1}$ are equal to one, indicating that all the relevant parameter intervals are sufficiently divided, and all the subspace angles are smaller than the maximum specified tolerance value.

7.2.3 Two parameters case

Similar to the single parameter case, equally spaced samples are chosen initially in both directions $\boldsymbol{\mu}_{i,j}$ $(i, j = 1, .., q-1)$ of a 2D parametric grid and the corresponding projection subspace bases $\mathbf{V}_{i,j}$ are obtained. Next, the distance between each diagonal sample pair is calculated, and the sum of angles between opposite sample pairs is recorded. This is to ensure that the parameter sensitivity of the model is captured in both the parametric directions. Next, this angle is checked against the maximum subspace angle and the following ratio is calculated

$$d_{i,j} = \frac{\theta_{(i,j),(i+1,j+1)} + \theta_{(i+1,j),(i,j+1)}}{\theta_{\max}}$$

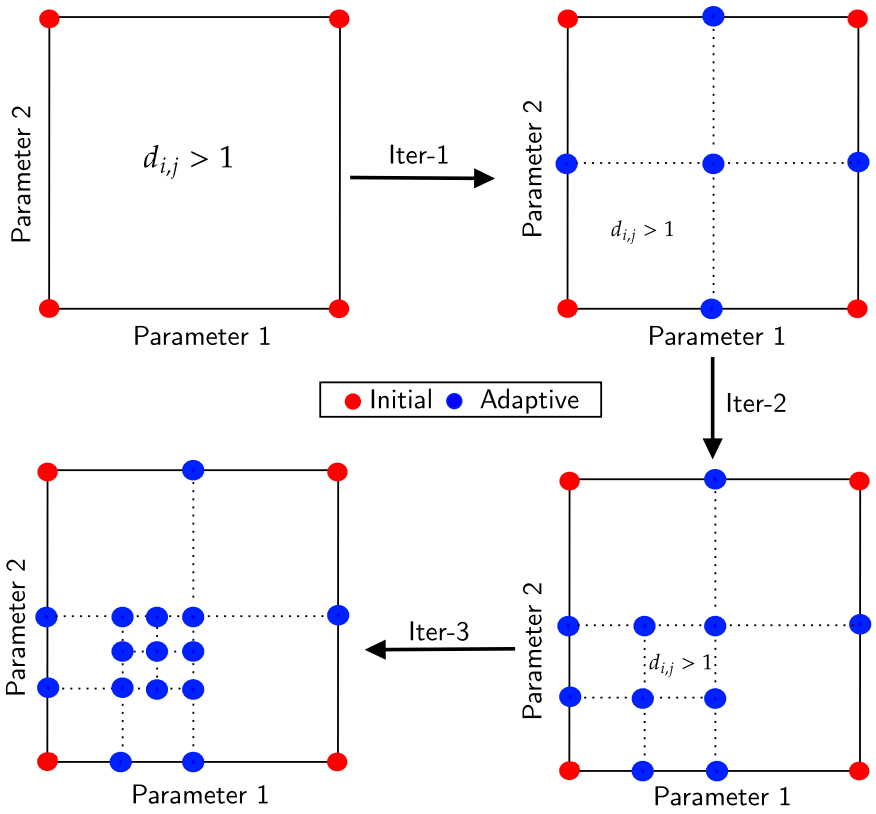

Figure 7.2: The adaptive sampling procedure demonstration for a 2-dimensional parametric grid

Similarly, a ceiling function of the ratio $d_{i,j}$ is used to resample the initial grid. If the ratio $d_{i,j}$ is larger than 1, the grid is divided into four small grids, which introduces five new samples (four at the midpoints of each side of the current grid and the fifth at the center of the grid). The process is repeated for the grids from the previous iteration, and the algorithm stops when all the ratios $d_{i,j}$ are equal to one. The adaptive scheme thus divides the initially selected grid into many smaller sub-grids, scanning each grid iteratively. This generates new samples in highly sensitive zones while the remaining samples are kept the same. Figure 7.2 depicts a scenario where three iterations are used to refine the initial grid adaptively, and Fig. 7.3 presents the flowchart of the proposed sampling scheme.

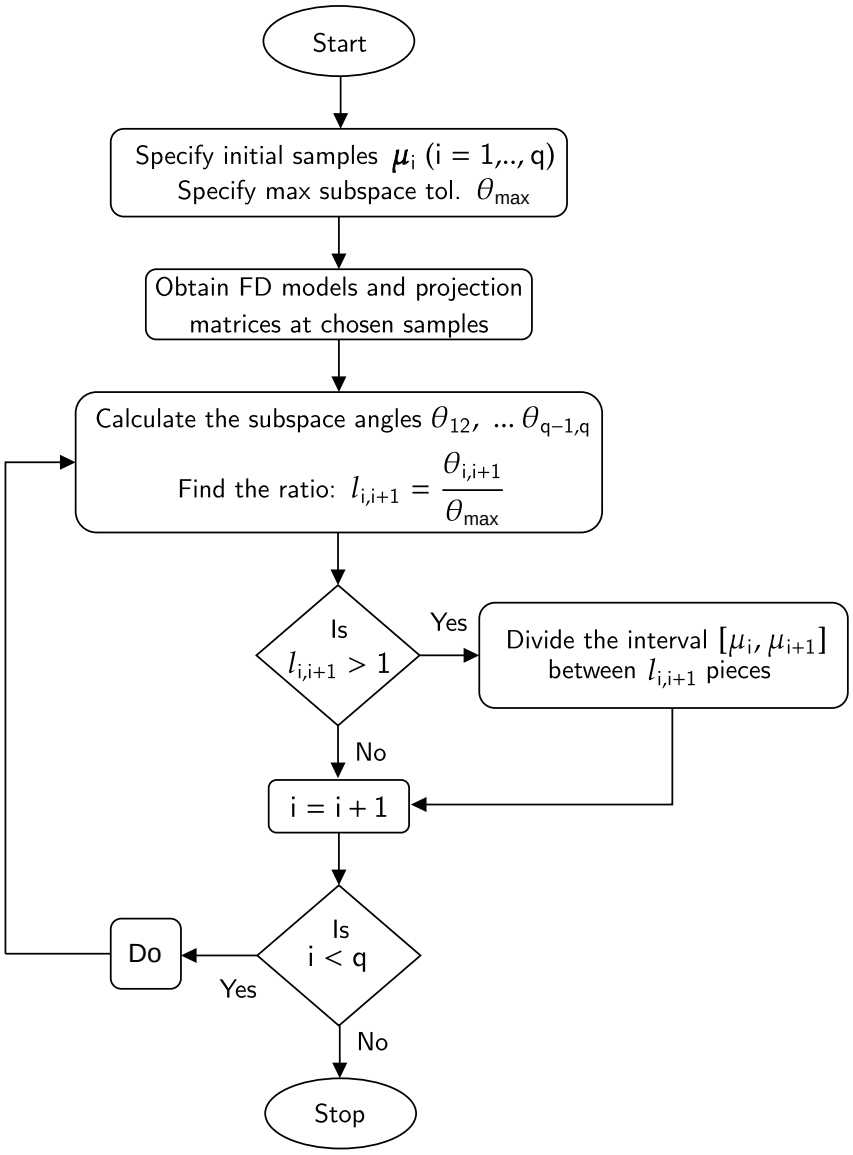

Figure 7.3: Automatic adaptive sampling scheme for matrix interpolatory pMOR

7.3 Numerical validation

This section delineates the proposed adaptive scheme on four nonlinear benchmark problems. The first two models have a single parameter dependency and the remaining models have two parameters.

7.3.1 Single parameter examples

1D Burgers' equation

We first consider the one-dimensional viscous Burgers' equation (cf. 6.10). The initial and boundary conditions are same as in Section 6.4. The various parameters used are enlisted in Table 7.1

Table 7.1: 1D Burgers' equation: Parameters of the selected scenarios

Method	Parameters
Full model	$n = 500, L = 1, t \in [0, 1.5s], h = 10^{-3}s, \boldsymbol{\mu} \in \mathcal{P} := [4 \times 10^{-4}, 4 \times 10^{-3}]$
ROM_d	NLMM: $r = 20, \dot{\boldsymbol{\zeta}}(t) = \tanh(t) + 5e^{-4}, i = 1, K = 101$ DEIM: $m_d = 10, n_s = 250$
Adaptive Sampling	$\theta_{\max} = 40^o$
Test input	$\mathbf{r}(t) = 0.5(\cos(2\pi/10)t)$

For our evaluations, we varied $\boldsymbol{\mu} \in \mathcal{P} := [4 \times 10^{-4}, 4 \times 10^{-3}]$. The output was taken as the average of state-vectors $\mathbf{x}$ and a single control input of $\mathbf{r}(t) = 0.5\cos((2\pi/10)t)$ was applied on left boundary. To test the proposed adaptive scheme, four FD models, each of size $n = 500$, were generated using implicit Euler's scheme with step-size of 10^{-3} for different viscosity values. This resulted in a uniform grid with four sample points for the initialization of the adaptive scheme. The local reduced-order models each of size $r = 20$, corresponding to each FD model, were constructed using the NLMM-DEIM (cf. Algorithm 5). The sample points μ_i and subspace angles for each parameter with its corresponding neighbor $\theta_{i,i+1}$ were recorded, and are enlisted in Table (7.2). It is seen clearly that the sensitivity of the model to parameter variations, as indicated by the subspace angles, increases as viscosity increases. This information is effectively captured by the subspace angles and thus clarifies the relative fitness of parametric mesh. For this model, the subspace angle tolerance was selected as $\theta_{\max} = 40^o$, and the parametric space was re-meshed for higher viscosity values wheres for low-viscosity values no refinement was needed. When the adaptive sampling algorithm finished, two new samples were obtained, i.e., $\mu = 2.2 \times 10^{-3}$ and $\mu = 3.4 \times 10^{-3}$ in the sensitive area (cf. Table 7.2), and the corresponding projection basis and the reduced models were stored. These ROMs were then transformed to the same set of coordinates.

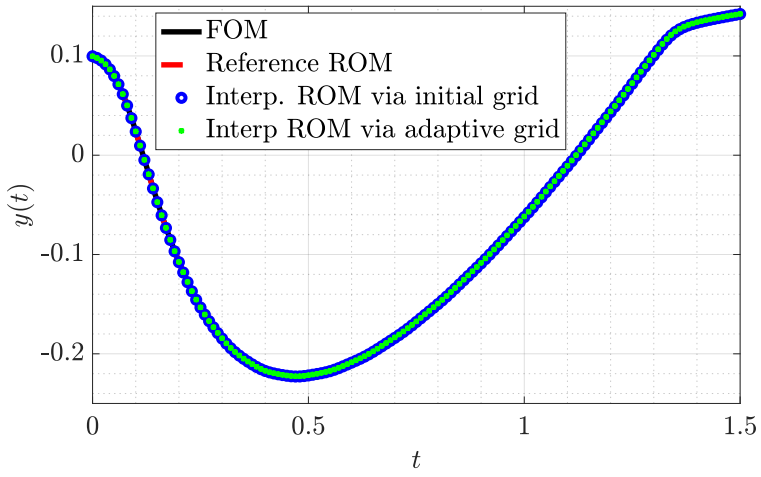

Figure 7.4: 1D Burgers' equation: Comparison of the output response for $\mu_{\text{test}} = 2 \times 10^{-3}$ (FOM size $n = 500$, ROMs size $r = 20$, DEIM indices $m_d = 20$)

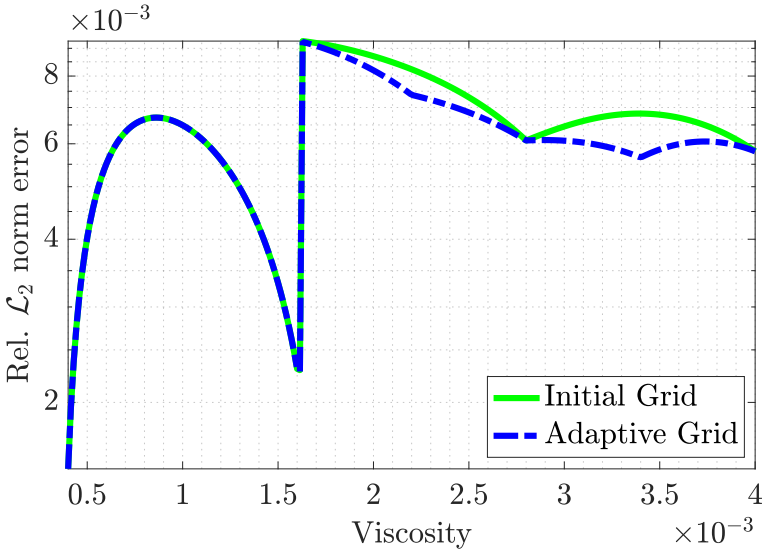

Figure 7.5: 1D Burgers' equation: Comparison of the relative $\mathcal{L}_2$ norm error between FOM and interpolated ROMs for different parameter

To highlight the advantages of the adaptive sampling scheme, a test parameter of value $\mu_{\text{test}} = 2 \times 10^{-3}$ (outside the training set) was selected randomly in the region of resampling and three different ROMs of sizes $r = 20$ were obtained. The first ROM was obtained via adaptive grid i.e., by interpolation of ROM 2 at $\mu = 1.6 \times 10^{-3}$ and ROM 3 at $\mu = 2.2 \times 10^{-3}$ (cf. Table 7.2). The second interpolated ROM was obtained from the initial grid via ROMs 2 and 4, respectively. The third ROM was directly obtained via the NLMM-DEIM procedure for reference. Figure 7.4 shows the output response of (6.10) for the chosen test parameter. It

Table 7.2: 1D Burgers' equation: Samples and subspace angles in the adaptive sampling scheme

Initialization	μ_i	0.0004		0.0016				0.0028			0.004
	$\theta^o_{i,i+1}$		36.92^o (1)			41.59^o (2)			46.37^o (2)		
Iter I	μ_i	0.0004		0.0016		0.0022		0.0028		0.0034	0.004
	$\theta^o_{i,i+1}$		36.92^o (1)		20.65^o (1)		20.94^o (1)		23.82^o (1)		22.55^o (1)

can be seen that the ROM via adaptive grid effectively captures the true response. Besides the qualitative comparison via output response, a quantitative analysis for 250 query points in the interval $\mu = [4\times10^{-4}, 4\times10^{-3}]$ was carried out. Figure 7.5 shows the relative $\mathcal{L}_2$ norm error between the FOMs and interpolated ROMs. As expected, the error is small in the proximity of grid points and large elsewhere due to interpolation error. The adaptive grid naturally yields small $\mathcal{L}_2$ norm errors than the initial grid for higher viscosity values.

MEMS microswitch

In the second test study, we examined the time-dependent, large-amplitude dynamics of a highly nonlinear squeeze-film damping problem involving electrostatic, mechanical and fluidic components [187, 215, 90, 311]. This nonlinear benchmark model is used as a pressure sensor due to its extreme sensitivity to surrounding atmospheric conditions [163, 195]. The switch consists of a polysilicon fixed-beam suspended over a polysilicon pad on a silicon substrate (cf. Fig. 7.6). An electrostatic force pulls the beam down to the pad if an external voltage is applied between the beam and the substrate. The dynamic behavior of this coupled electromechanical system can be modeled with 1D Euler's beam equation and 2D Reynolds squeeze film damping equation given as:

$$\hat{E}I_0h^3w\frac{\partial^4 z}{\partial x^4} - S_0hw\frac{\partial^2 z}{\partial x^2} = F_e + \int_0^w (p - p_a dy) - \rho_0 hw\frac{\partial^2 z}{\partial t^2} \tag{7.2a}$$

$$\nabla.[(1+6K)z^3p\nabla p] = 12\mu\frac{\partial(pz)}{\partial t} \tag{7.2b}$$

where the electrostatic force across the plates is given as $F_e = -(\epsilon_0 wv^2/(wz^2))$ which is due to applied input voltage $u(t) = (3cos(2\omega t)+7cos(0.5\omega t))^2$ with $\omega = 2\pi$ Ghz. The measured output is the height of the beam z. The length of beam is $l = 610\mu$m whereas a width is $w = 40\mu$m. ϵ_0 and μ represent the permittivity and permeability respectively. $I_0 = 1/12$ is the moment of inertia, $\hat{E} = 149$GPA represents the Young's modulus, $K = 0.064/z_0$ is the Knudsen number, $S_0 = -3.7$ represents the stress coefficient and $\rho_0 = 2300kg^3$ is the density. By assuming the

state variables $\mathbf{x}_1 = z$, $\mathbf{x}_2 = \partial \mathbf{x}^3/\partial t$ and $\mathbf{x}_3 = p$ where p is the pressure, the finite discretization scheme with N steps along the length of the beam and M steps along the width resulted in the following state-space model of dimension $(2+M)N$:

$$\begin{aligned}
\frac{d\mathbf{x}_1}{dt} =& \frac{\mathbf{x}_2}{3\mathbf{x}_1^2} \\
\frac{d\mathbf{x}_2}{dt} =& \frac{2\mathbf{x}_2^2}{3\mathbf{x}_1^3} + \frac{3\mathbf{x}_1^2}{\rho_0 hw}\left[\int_0^w (\mathbf{x}_3 - p_a)dy + S_0 hw\frac{\partial^2 \mathbf{x}_1}{\partial \mathbf{x}_2}\right. \\
& \left. - EIh^3 w\frac{\partial^4 \mathbf{x}_1}{\partial \mathbf{x}^4}\right] - \frac{3\epsilon_0}{2\rho_0 h}u(t) \\
\frac{d\mathbf{x}_3}{dt} =& -\frac{\mathbf{x}_2\mathbf{x}_3}{3\mathbf{x}_1^3} + \frac{1}{12\mu\mathbf{x}_1}\nabla\left[(1+6\frac{\lambda}{\mathbf{x}_1})\mathbf{x}_1^3\mathbf{x}_3\nabla\mathbf{x}_3\right]
\end{aligned} \tag{7.3}$$

where the boundary conditions are given as:

$$\begin{aligned}
&\mathbf{z}(\mathbf{x},0) = 2.3\mu m, && \mathbf{p}(\mathbf{x},\mathbf{y},0) = 1.103\times 10^5 \text{ Pa}, \\
&\mathbf{z}(0,t) = \mathbf{z}(l,t) = 2.3\mu m, && \frac{\partial \mathbf{p}(0,\mathbf{y},t)}{d\mathbf{x}} = \frac{\partial \mathbf{p}(l,\mathbf{y},t)}{\partial \mathbf{x}} = 0 \\
&\mathbf{p}(\mathbf{x},0,t) = \mathbf{p}(\mathbf{x},\mathbf{w},t) = 1.103\times 10^5 \text{ Pa}.
\end{aligned}$$

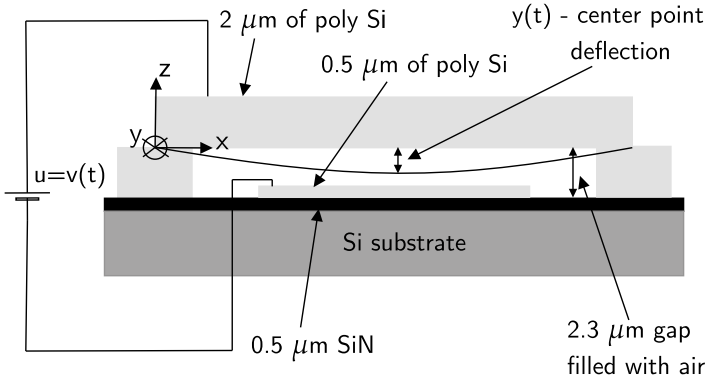

Figure 7.6: The MEMS switch

For this model, the beam width was selected as the running parameter. We selected a uniform initial grid of four sample points for varying beam widths $\boldsymbol{\mu} \in \mathcal{P} := [45, 75]$ with an equal span of $10\mu m$. As such, four different FD models, each of dimension $n = 450$, were obtained. The reduced-order models of size ($r = 25$), corresponding to each full model, were constructed using 500 POD snapshots obtained via implicit Euler's scheme with a step size of 10^{-6}. A subspace tolerance of $\theta_{\max} = 10^o$ was selected for the adaptive sampling scheme. At the end

Table 7.3: MEMS microswitch model: Samples and subspace angles in the adaptive sampling scheme

Initialization	μ_i	45		55			65				75
	$\theta^o_{i,i+1}$		5.28^o			17.80^o (2)			11.59^o (2)		
Iter I	μ_i	45		55		60		65		70	75
	$\theta^o_{i,i+1}$		5.28^o		8.27^o		9.59^o		6.04^o	5.47^o	

of the adaptive scheme, the grid was remeshed, and two new sample points were introduced, i.e., at $w = 60\mu m$ and $w = 70\mu m$. The subspace angles and the final sample points can be seen in Table 7.3. The ROMs were then transferred to the same set of coordinates, and a universal subspace was constructed, as explained in the previous section.

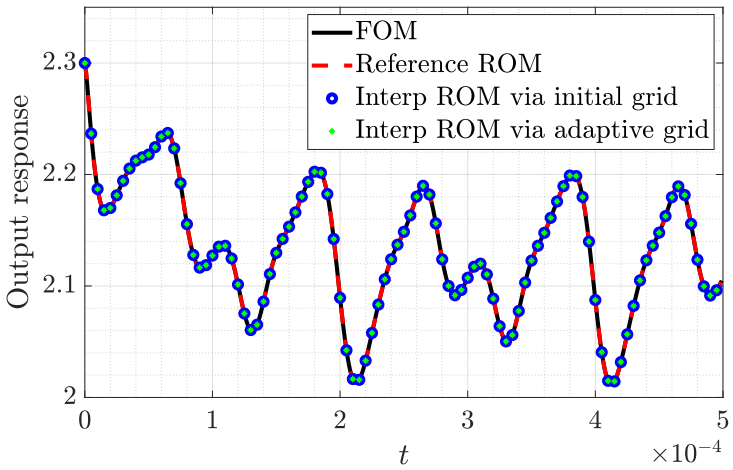

Figure 7.7: MEMS microswitch model: Comparison of the output response of for test parameter $w_{\text{test}} = 73\mu m$ FOM size $n = 450$, ROMs size $r = 25$

To proceed with the online testing, a test parameter of value $w_{\text{test}} = 73\ \mu m$ was randomly selected. The interpolated ROM was then obtained from neighboring models for parameter values $w = 70\ \mu m$ and $w = 75\ \mu m$ resp. Similar to the previous model, the interpolated ROM via adaptive grid was compared with the ROM obtained from the initial grid as shown in Fig. 7.7. The quantitative comparison of the ROMs for 250 test samples in the same parametric span is presented in Fig. 7.8. It is clearly seen that ROMs obtained through adaptive grid yield better approximations than obtained via the initial grid for increasing switch widths.

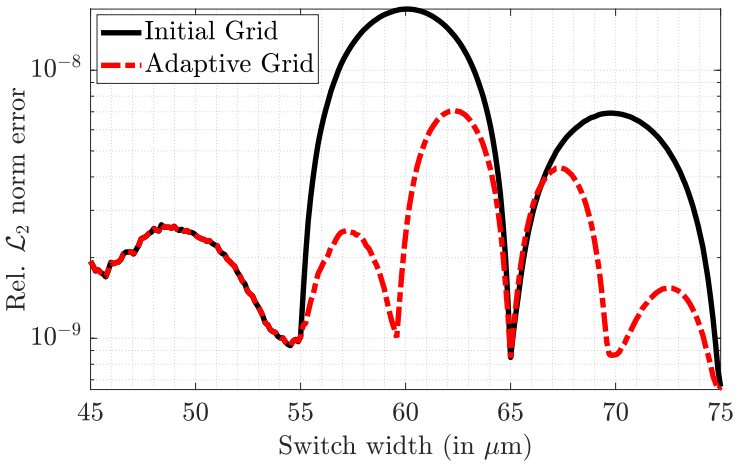

Figure 7.8: MEMS microswitch model: Comparison of the relative $\mathcal{L}_2$ norm error between FOM and the interpolated ROMs

7.3.2 Two-parameter examples

1D Reaction-Diffusion equation

Next, we considered the one-dimensional Reaction-Diffusion equation also known as the *Chafee-Infante* equation.

$$\frac{\partial \vartheta}{\partial t}(\mathbf{x},t) = \delta \frac{\partial^2 \vartheta}{\partial \mathbf{x}^2}(\mathbf{x},t) - \lambda(\vartheta^3(\mathbf{x},t) - \vartheta(\mathbf{x},t)), \tag{7.4}$$

with initial condition and boundary conditions as:

$$\begin{aligned} &\vartheta(\mathbf{x},0) = 1 - \sin(\pi \mathbf{x}/L), && \mathbf{x} \in (0,L), \\ &\vartheta(0,t) = \mathbf{u}(t), \qquad \frac{\partial \vartheta}{\partial \mathbf{x}}(L,t) = 0 && t \in (0,T). \end{aligned}$$

Here, λ and δ are two parameters of interest. The various parameter used during the simulations are mentioned in Table 7.4

Table 7.4: 1D Reaction-Diffusion equation: Parameters used

Method	Parameters
Full model	$n = 1000, L = 1, \lambda = 1, t \in [0, 3s], \delta = [0.1 \times 0.3], \lambda = [1 \times 2]$
NLMM	$r = 30, \dot{\boldsymbol{\zeta}}(t) = 0.5e^{-t/5} + \pi\cos(\pi t)e^{-t/5} + \sin(\pi t)e^{-t/5}, i = 1, K = 61,$ $t_k \in [0, 10s]$
DEIM	$m_d = 50, n_s = 200$
Adaptive Sampling	$\theta_{\max} = 100^o$
Test Input	$\mathbf{u}(t) = 0.5(\cos(\pi t) + 1)$

The PDE was discretized in space for $n = 1000$ spacial grid points using the FDM, and the output of interest was measured on the right boundary. With a diffusion-dominated regime ($\lambda > \delta$), nine FD models were initially obtained at an equidistant sample spacing on a 2D parametric grid as shown in Fig. 7.9. The high-fidelity models were solved via implicit Euler's scheme with a uniform step size of 0.01. The reduced models of size $r = 30$ were constructed using the NLMM procedure (via Algorithm 4), and the nonlinearity was efficiently evaluated using $m_d = 50$ DEIM indices (via Algorithm 3). The subspace angles for this two-parameter system were measured between each diagonal pair. The tolerance for maximum subspace angle was chosen as $\theta_{\max} = 100^o$, and the grid was automatically remeshed whenever the sum of the distance between opposite pairs exceeded the tolerance value as explained in the previous section. At the end of the offline adaptive sampling scheme, thirteen new sample points were added to the previously selected grid as indicated by blue dots on the 2D map shown in Fig. 7.9.

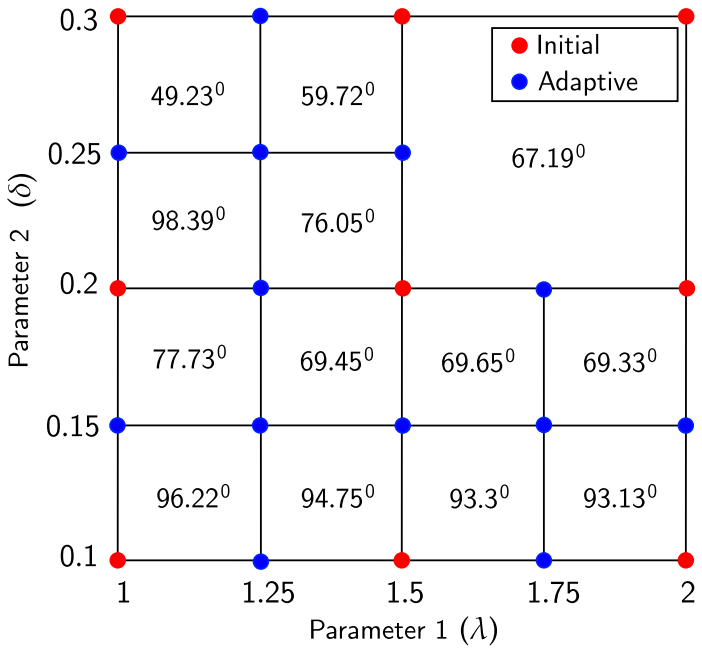

Figure 7.9: 1D Reaction-Diffusion model: Initial and adaptive grid samples at the end of adaptive sampling scheme. The sum of subspace angles of the diagonal pairs is also mentioned

To validate the proposed adaptive scheme for a two-parameter system, a test parameter value of ($\lambda_{\text{test}} = 1.244$, $\delta_{\text{test}} = 0.105$) was randomly selected, and the interpolated ROM was obtained from the adaptive grid and compared with ROM via the initial grid. A directly reduced model via NLMM-DEIM was also obtained at the same test sample. Fig. 7.10 shows the qualitative comparison of the output response for the full model and the reduced models. It is observed that the true

response of the FOM was accurately replicated by both the interpolated ROMs. However, the approximating quality of the interpolated ROM obtained via the adaptive sampling scheme is more than the ROM obtained by regular sampling. This is demonstrated in Fig. 7.11 where the relative $\mathcal{L}_2$-norm errors between FOM and the different interpolated ROMs are plotted for 100 different query points in parametric region. The ROMs obtained via adaptive grid naturally yields a better approximation since these values were adaptively refined.

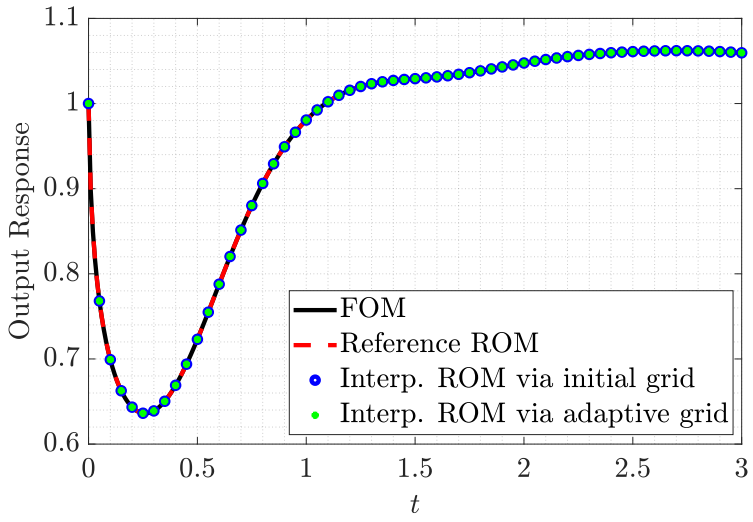

Figure 7.10: 1D Reaction-Diffusion model: Comparison of the output response for test parameters $\lambda_{\text{test}} = 1.244, \delta_{\text{test}} = 0.105$

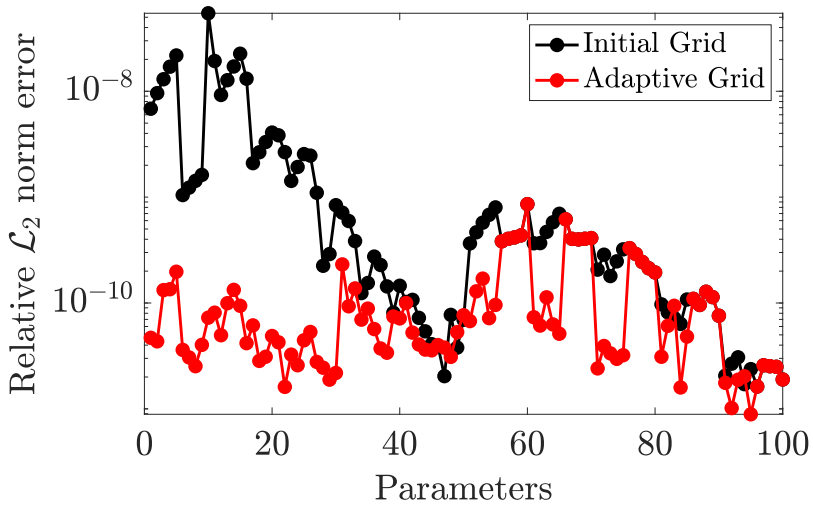

Figure 7.11: 1D Reaction-Diffusion model: Comparison of the relative $\mathcal{L}_2$ norm error between FOM and interpolated ROMs

Nonlinear RC ladder

Finally, we considered the nonlinear analog RC circuit example from Ref. [61] to illustrate the adaptive sampling scheme. The circuit, given in Fig. 7.12, has a chain of strong nonlinear diodes together with resistors and capacitors. This model is a variant of transmission line model discussed in Sec. 4.3.3. Kirchhoff's nodal law and current equations were used to derive the state equations for this system. An equation for i^{th} interior node is given as:

$$\begin{aligned} C\frac{dv_i}{dt} =& \frac{1}{R}(v_{i-1} - v_i) - \frac{1}{R}(v_i - v_{i+1}) \\ &+ I_d\left[e^{\alpha(v_{i-1}-v_i)} - e^{\alpha(v_i-v_{i+1})}\right]. \end{aligned} \tag{7.5}$$

The nodal voltages are selected as the state-variables $x(t) = [v_1\ v_2\ ...\ v_N]^T$. The input is the current source $u(t) = i(t)$ and the output is voltage measured at the first node. The state-space model is given as:

$$\mathbf{E}\frac{d\mathbf{x}}{dt} = \mathbf{G}\mathbf{x}(t) + (\mathbf{D}\mathbf{x}(t)) + \mathbf{B}\mathbf{u}(t), \tag{7.6}$$

where $\mathbf{E}$ is the capacitance matrix, $\mathbf{G}$ is the conductance matrix, $\mathbf{D}(\mathbf{x})$ is a vector valued nonlinear function for the diodes and $\mathbf{B} = [1\ 0...0]^T$ is the input vector. The various parameters of the model and reduction are mentioned in Table 7.5)

Table 7.5: Nonlinear Transmission Line: Parameters used

Method	Parameters
Full model	$n = 1000, R = C = 1, t \in [0, 10s]$,$\boldsymbol{\mu_1}, \boldsymbol{\mu_2} \in \mathcal{P} := [40, 43] \times [0.8, 1]$
POD	$r = 15, n_s = 250$
DEIM	$m_d = 30$
Adaptive Sampling	$\theta_{\max} = 100^o$
Test Input	$\mathbf{u}(t) = 0.5(1 + \cos(\pi t)/5)$

The constitutive relation for diodes is given as:

$$i_d(v) = I_d(e^{\alpha v} - 1), \tag{7.7}$$

with $\alpha = 1/v_t$ and v_t is the threshold voltage. Corresponding to $v_t = 25mV$, the nominal values for diodes is $I_d = 0.1nA$ and $\alpha = 40$. The nonlinear analog circuit was parameterized both in I_d and α with a range $\boldsymbol{\mu_1}, \boldsymbol{\mu_2} \in \mathcal{P} := [40, 43] \times [0.8, 1]$. Like the previous test model, nine initial test samples were selected with a uniform spacing, shown by red dots in Fig. 7.13, and corresponding to each sample, an expensive high-fidelity model of size $n = 1000$ was obtained. The reduced-order models of size ($r = 15$) were constructed with 250 time-snapshots within a POD

scheme, whereas the nonlinearity was captured using $m_d = 30$ DEIM indices. The maximum subspace tolerance was chosen as $\theta_{\max} = 100^o$ for initialization of the adaptive sampling scheme. Fig. 7.13 shows the final adaptive grid for the nonlinear RC circuit along-with the sum of subspace angles of the diagonal pairs. The initial grid was adaptively re-meshed based on the sensitivity of the parametric variations captured in terms of the distance between the subspaces. The adaptive scheme terminated when all the subspace angles between the neighboring samples were smaller than the desired set tolerance value.

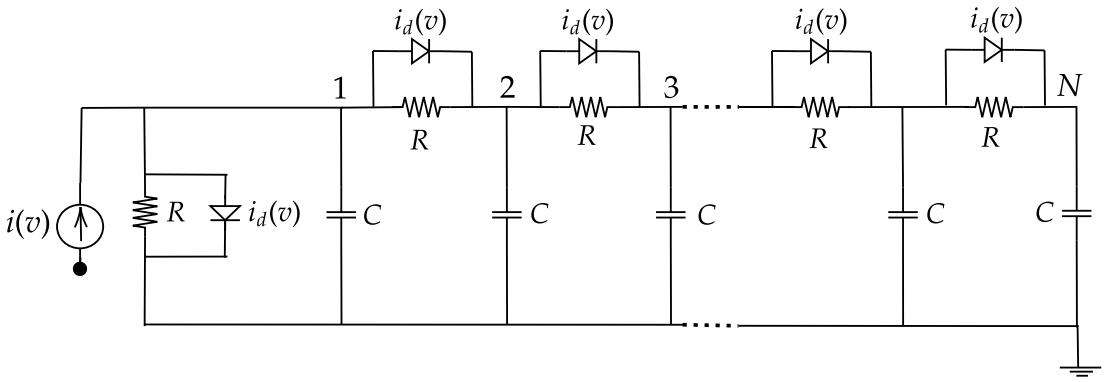

Figure 7.12: Nonlinear RC ladder network

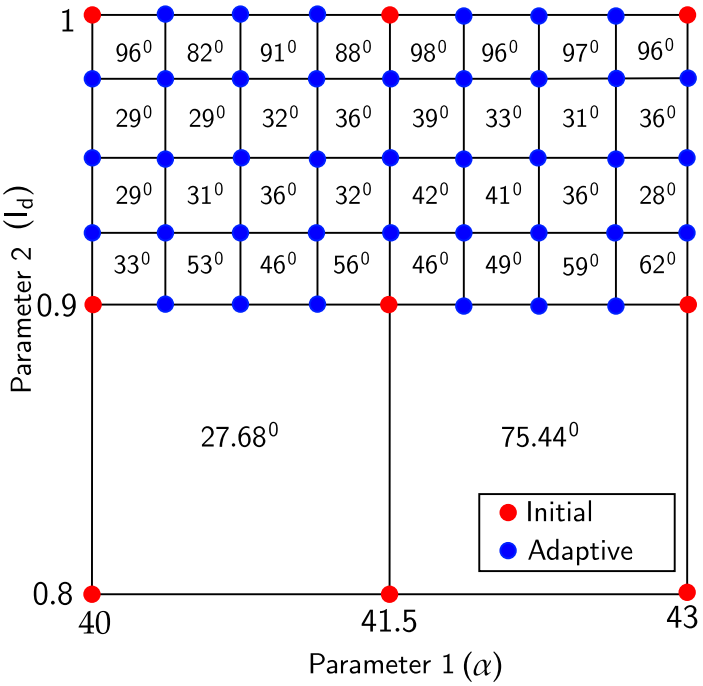

Figure 7.13: Nonlinear RC circuit: Initial and adaptive grid samples. The sum of subspace angles of the diagonal pairs is also mentioned

The online testing was performed for a test parameter of value $\alpha_{\text{test}} = 40.10$ and $I_{d_{\text{test}}} = 0.911nA$ corresponding to the region where the adaptive meshing was performed. As such, the interpolated ROMs corresponding to both initial and final grid were obtained along-with the directly-reduced model via POD-DEIM. The comparison of the output responses of the ROMs against the full-order model is shown in Fig. 7.14. It is clearly seen that the interpolated ROM via final grid effectively captures the response with a high level of accuracy just like the directly obtained reduced model. On the other hand, the interpolated ROM obtained from the initial grid produced large errors due to inefficient sampling. Moreover, the quantitative comparison for 100 different test parameters in terms of the relative $\mathcal{L}_2$ norm errors with respect to FOM is presented in Fig. 7.15. It is seen that the $\mathcal{L}_2$ error or ROMs obtained from adaptive grid had dropped by several orders of magnitude.

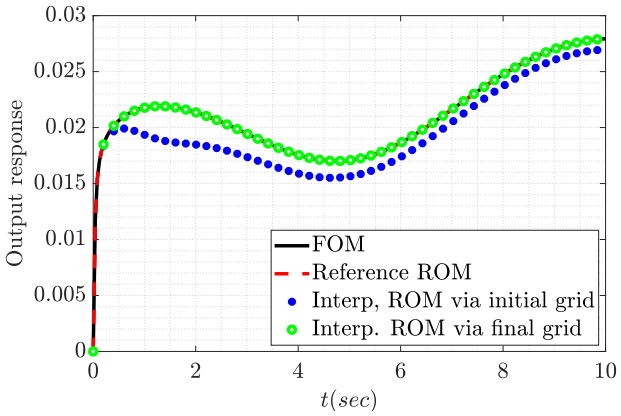

Figure 7.14: Nonlinear RC circuit:Comparison of the output response for $\alpha_{\text{test}} = 40.10, I_{d_{\text{test}}} = 0.911nA$. FOM size: $n = 1000$, ROM size: $r = 15$, number of DEIM indices = 30

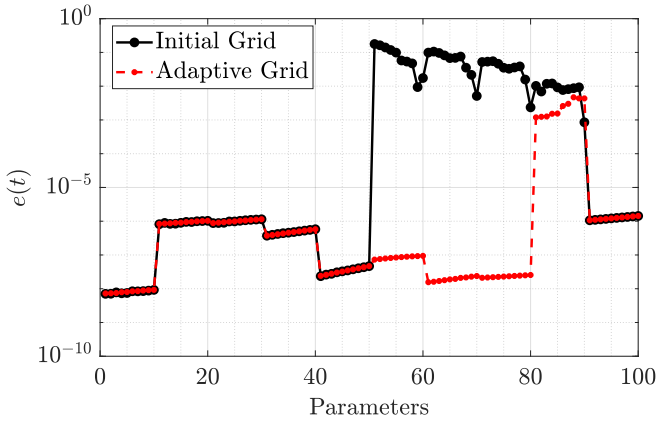

Figure 7.15: Nonlinear RC circuit: Comparison of the relative $\mathcal{L}_2$ norm error between FOM and interpolated ROMs

7.4 Discussions

In this section, we discuss some of the aspects the proposed sampling scheme.

Advantages of the proposed scheme

The automatic adaptive sampling scheme for nonlinear systems described herein avoids the computation of a new FD model every time a new parameter is introduced. The models are being calculated only at those parameters where the dynamic response changes significantly, measured by observing the change in attributes of projection subspaces. The subspace angles are continuously tracked and are refined automatically whenever they exceed a pre-defined tolerance ensuring that the same FD model is used. It should be noted that the whole process is performed in the offline stage before the matrix interpolation in the online stage begins.

Since this scheme uses the information from the system itself to resample the parametric domain; it can be used to reduce the substantial offline computational costs associated when a finely sampled training set is used in the reduced basis method. A course training set can be first resampled based on the proposed scheme and then through a greedy algorithm to obtain the approximated solution manifold of the original system as carried out by Benner et al. [52] for optimization of batch chromatography.

Selection of subspace tolerance angle $\theta_{\max}$

The selection of the maximum subspace tolerance $\theta_{\max}$ is problem-specific in nature. A small tolerance value will result in a finer mesh and hence a small interpolation error. However, this also increases the associated offline computational time as more FD models will be involved. On the other hand, a larger tolerance value will result in a few FD models with increased interpolation errors. Thus, depending upon the application and the nonlinear model at hand, the value of $\theta_{\max}$ can be chosen.

Choice of reduction scheme used and size of ROM

The proposed sampling scheme is independent of the choice of reduction scheme involved, which has been validated in this study using different reduction approaches. However, we advocate using the NLMM method as it avoids the need to obtain the time-snapshots of the FOM and instead constructs the local projection basis in a "simulation-free" environment. Furthermore, it is advised to use hyperreduction schemes like DEIM, EIM, GNAT, etc., to evaluate the underlying nonlinearity efficiently.

More accurate FD models and larger ROMs can more accurately capture the dynamics and improve the interpolated ROM quality. However, this will also result

in the enormous offline computation of the projection basis. Methods like SVD can be used to calculate dominant singular values that capture the most variance of the high-fidelity model and provide an estimate for appropriate truncation of the local basis.

Computational efficiency

The efficiency of the proposed scheme was demonstrated in terms of error norms in this study. Since the adaptive method works in an offline-online manner, the efficiency can also be shown in terms of overall CPU times. Although the offline cost is usually not taken into consideration for a PMOR scheme, it is typically high, especially for time-dependent PDEs. The CPU times recorded for all tests are presented in Table 7.6 for reference.

Table 7.6: CPU times for different test models

S.No.	Model	offline time (s)	online time (s)
1.	Burgers' equation	911.23	4.12
2.	MEMS microswitch	1148.75	8.31
3.	Reaction-Diffusion equation	1225.79	13.56
4.	Nonlinear RC network	956.31	7.76

The offline time includes the time required to obtain the FD models at selected samples along-with obtaining local projection basis. Online time includes the interpolation time of the ROMs.

Curse of dimensionality for higher dimension spaces

The proposed scheme can be extended to higher-dimensional parameter setup. However, it becomes more involved as it may require a considerable adaptation since full grids become computationally infeasible in such a scenario. As such, adaptive sparse grids can be employed [229]. Currently, the proposed adaptive scheme can be used for parameter dimensions up to two.

Limitations in the proposed sampling scheme

The proposed scheme currently has certain limitations such as stability preservation, lack of global error bounds, infeasibility for a much higher dimensional parametric space. These limitations can be improved in a future research.

7.5 Chapter Summary

This chapter studied the adaptive parameter sampling scheme for nonlinear systems. We presented an overview of some notable sampling schemes and discussed their shortcoming. Then, we introduced an adaptive sampling strategy for nonlinear systems based on the system-theoretic measures. We combined the matrix-interpolatory approach for nonlinear systems with adaptively choosing the sampling points for obtaining local reduced models. The distance between the subspaces was used to identify the most sensitive zones of the parametric space. Simultaneously, the whole process was automated to directly obtain a reduced model for a new parametric variation outside the training set. The algorithm was tested on various benchmark nonlinear systems showing satisfactory results. The proposed scheme is beneficial, especially where parameter dependency in the model is implicit, like the finite element method.

Part IV

Closure and Appendices

Chapter 8

Concluding Remarks

The idea of automatically extracting reduced-order models from large-scale dynamical systems is a central challenge. By and large, reduced-order modeling via moment-matching techniques for linear systems is relatively well explored, and a suite of efficient algorithms is available at our disposal. However, constructing optimal approximations of the large-scale models while preserving passivity or stability remains elusive. Furthermore, technical systems arising while modeling complex fluids or simulation of electronics circuits are inherently nonlinear. Thus employing nonlinear reduction techniques directly to these systems is desirable. The nonlinear model order reduction area, though witnessed some notable achievements in the past few years, however some issues (as discussed in this thesis) remains challenging. Moment-matching-based MOR methods have seen a long history. Various modifications and advancements have appeared from time to time to make this technique a more robust platform for treating problems in large-scale settings. Some of these modifications are currently a work in progress, whereas others are yet to happen.

In essence, this thesis examines the problem of nonlinear model order reduction using system-theoretic measures. In this respect, we have proposed some efficient reduction frameworks for nonlinear state-space models of time-dependent PDEs. Notably, we show how NLMM, and its variants, can be used to obtain a "snapshot-free" ROM architecture undergoing implicit moment-matching with the original high-dimensional system. Several numerical simulations are presented to realize the proposed observations. Then we also discussed the extension of these techniques to systems in second-order state-space form. In the realm of nonlinear parametric state-space models, we have proposed a numerical framework for parametric reduction using the matrix interpolation method. We have also introduced a sampling strategy to enhance the approximating qualities of parametric ROMs.

In this chapter, we present a summary of the overall contributions of the thesis, draw conclusions from the most important results, and discuss the scope for future work.

8.1 Summary and conclusions

Chapter 2

1. The problem of MOR in linear systems has reached a considerable level of maturity, as reflected by several books and monographs.

2. Moment-matching-based MOR methods have emerged as a vibrant and most broadly used approach to treat models of large-scale nature. This class of methods is based on Krylov subspace methods and is suitable for a wide range of engineering applications.

3. The success of interpolatory methods lies in the fact that a few matrix-vector multiplications are involved, and the complexity of resulting models is roughly $\mathcal{O}(nk^2)$ for a sparse matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ where k is the number of expansion points.

4. Classical frequency-domain interpretation of moment-matching corresponds to interpolating the transfer functions of FOM and ROM at specific shifts, where the time-domain interpretation of moment-matching corresponds to the interpolation of the steady-state responses of the FOM and ROM.

5. Interconnecting a linear signal generator with FOM corresponds to exciting the FOM with exponential inputs. Using a nonlinear signal generator corresponds to exciting the FOM with user-defined inputs.

Chapter 3

1. The method of using linear projection for reducing nonlinear systems is well understood. Famous methods like POD, TPWL, quadratic methods perform satisfactorily for a large number of nonlinear systems. However, care must be taken while selecting the training inputs, linearization points, and temporal snapshots.

2. Using a nonlinear Petrov-Galerkin projection offers a more accurate approximation of the lower-dimensional subspace; however, this complicates the structure of the ROM.

3. Classical hyper-reduction methods like EIM, or POD-tailored DEIM, gappy POD, etc., can be used "on-top-of" the usual reduction methods for obtaining nearly optimal reconstructions of nonlinear terms of the original high-dimensional system.

Chapter 4

1. Nonlinear moment-matching has emerged as a promising system-theoretic reduction technique for nonlinear systems since it doesn't involve expensive FOM simulations but solves the underlying nonlinear Sylvester PDE to obtain the projection basis.

2. Since the nonlinear Sylvester PDE is expensive to simulate, an approximated version is solved instead using some numerical simplifications.

3. We have used NLMM with DEIM to obtain a compact ROM structure. This framework is demonstrated on several benchmark examples and compared with simulation-based methods like POD.

4. We have also used NLMM with DMD to obtain a non-intrusive reduction framework. This technique produces a higher computational savings than POD-DEIM which has been demonstrated through several examples.

Chapter 5

1. Models emerging from mechanical and electrical systems usually have a second-order structure, and, as such structure-preserving ROMs are highly desired.

2. Towards this direction, we have extended the NLMM-DEIM framework to second-order systems to obtain reduced models that preserve the structure of the original model.

3. SO-NLMM-DEIM has been implemented on power system models, and the resulting reduced models satisfactorily capture the strong nonlinear transients of the original model.

Chapter 6

1. Many technical systems include parameters that may be considered as deterministic or stochastic variables of interest.

2. The goal of PMOR approaches is to obtain a reduced model approximating the response of the full-order system with high fidelity over a wide range of parametric variations.

3. In this context, we have presented an offline-online PMOR framework using the matrix interpolatory scheme for nonlinear systems. The method has been tested on several nonlinear models with parametric dependencies.

Chapter 7

1. One of the central challenges in PMOR is parametric sampling. The need for pMOR methods with automatic parameter sampling is strongly aimed. Methods like Greedy sampling [234], POD inspired methods [200, 136] or the rational interpolation-based methods [112] are the popular choices. However, more sophisticated sampling approaches such as problem-aware adaptive search are mandated as parametric space grows.

2. Using the notion of distance between subspaces, an adaptive sampling method for PMOR has been proposed. Numerical simulations have been carried out to test the proposed scheme for a single and two-parameter setup. The results show that the adaptive sampling has significantly increased the approximating qualities of the parametric ROMs.

8.2 Future perspectives

Several other perspectives have already been mentioned throughout the thesis. In what follows, we enlist some promising directions for possible future research.

1. The idea of output krylov subspace based NLMM as proposed in Maria et al. [88] can be further pursued to obtain a two-sided reduction method. Some critical system properties such as stability and passivity can be examined from a theoretical and numerical perspective.

2. The parameterized families of ROMs achieving moment-matching mentioned in Chapter 3 can be further exploited to enforce specific additional properties such as matching asymptotic stability, matching prescribed relative degree, passivity constraints.

3. This thesis uses reduction methods that project the governing equations onto a linear subspace approximation of the original state-space. This restricts the states to evolve in a linear subspace and hence imposes a limitation on the accuracy of the resulting ROM. This effect is more pronounced for problems with slow decaying Kolmogorov n-width, such as in advection-dominated regimes. It is recommended to project the underlying dynamical system onto nonlinear manifolds for such problems. To overcome the Komgolorov width of the problem, deep convolutional autoencoders to learn the trial manifolds can be an exciting research direction.

4. Dynamic equivalencing in power systems is a promising area of research. With the anticipated transition to smart-grids, there will be additional fluctuations in the system from new components such as plug-in electric vehicles, intermittent energy sources, etc. Such fluctuations will have an impact on the synchronization stability of power grids and instabilities caused by the

so-called dynamic bifurcation problems [57]. Though some notable reduction techniques for power systems have been proposed in the past based on Krylov subspace [78, 262], balanced truncation [245] and extended balanced truncation [280], however, some open problems remain unexplored. This includes preservation of physical meanings of state variables, model-free approaches for prediction, and developing software implementation in proprietary and dedicated power system analysis tools.

5. Adaptive parametric sampling from Chapter 7 can be further extended to a higher dimension of parametric space to test more practical systems arising from all engineering domains.

Appendix A

Detailed description of the nonlinear models

A.1 1D Nonlinear Burgers' equation

$$\mathbf{A} = \begin{bmatrix} a & b & 0 & \cdots & 0 \\ b & a & b & \cdots & 0 \\ 0 & b & a & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & b \\ 0 & 0 & \cdots & b & c \end{bmatrix} \in \mathbb{R}^{n\times n},\ \mathbf{B} = \begin{bmatrix} b \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \in \mathbb{R}^{n\times 1},\ \mathbf{N} = \begin{bmatrix} f & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{n\times n}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & \cdots & 1 \end{bmatrix} \in \mathbb{R}^{1\times n},\ \mathbf{H} = \begin{bmatrix} d & 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & e & 0 & \cdots & 0 & e & d & \cdots & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & e & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & \cdots & \cdots & d \end{bmatrix} \in \mathbb{R}^{n\times n^2}$$

where

$$a = \frac{-2\nu}{h^2}, b = \frac{\nu}{h^2}, c = \frac{-\nu}{h^2}, d = \frac{-1}{h}, e = \frac{1}{2h}, f = \frac{1}{h}, h = \frac{1}{1+n}, \nu = \frac{1}{\mathrm{Re}}, n = \text{order of the system}$$

A.2 2D Nonlinear Burgers' equation

The spatial domain is discretized uniformly into $n_x - 1$ and $n_y - 1$ intervals in x and y direction, respectively for a $n_x \times n_y$ grid. Now, using a centered-difference scheme for the first and second-order spatial derivatives, we arrive at the following

equations:

$$\frac{d\mathbf{u}_{i,j}}{dt}+\frac{\mathbf{u}_{i+1,j}-\mathbf{u}_{i-1,j}}{2d\mathbf{x}}\mathbf{u}_{i,j}+\frac{\mathbf{u}_{i,j+1}-\mathbf{u}_{i,j-1}}{2d\mathbf{y}}\mathbf{v}_{i,j}= \frac{1}{Red\mathbf{x}^2}(\mathbf{u}_{i+1,j}-2\mathbf{u}_{i,j}+\mathbf{u}_{i-1,j})+\frac{1}{Red\mathbf{y}^2}(\mathbf{u}_{i,j+1}-2\mathbf{u}_{i,j}+\mathbf{u}_{i,j-1}) \tag{A.1a}$$

$$\frac{d\mathbf{v}_{i,j}}{dt}+\frac{\mathbf{v}_{i+1,j}-\mathbf{v}_{i-1,j}}{2d\mathbf{x}}\mathbf{u}_{i,j}+\frac{\mathbf{u}_{i,j+1}-\mathbf{u}_{i,j-1}}{2d\mathbf{y}}\mathbf{v}_{i,j}= \frac{1}{Red\mathbf{x}^2}(\mathbf{v}_{i+1,j}-2\mathbf{v}_{i,j}+\mathbf{v}_{i-1,j})+\frac{1}{Red\mathbf{y}^2}(\mathbf{v}_{i,j+1}-2\mathbf{v}_{i,j}+\mathbf{v}_{i,j-1}) \tag{A.1b}$$

where

$$\mathbf{x}_j = jd\mathbf{x}, \mathbf{y}_i = id\mathbf{y}, \mathbf{u}_{ij} = \mathbf{u}_{x_i,y_j,t}, \mathbf{v}_{ij} = \mathbf{v}(\mathbf{x}_i, \mathbf{y}_j, t), d\mathbf{x} = \frac{b-a}{n_x-1}, d\mathbf{y} = \frac{d-c}{n_y-1}.$$

A more compact notation for Equation (A.1) is given as:

$$\frac{d\mathbf{U}}{dt}+\mathbf{f}_1(\mathbf{U},\mathbf{V})-\frac{1}{2d\mathbf{x}}\mathbf{B}_{ul}\mathbf{U}-\frac{1}{2d\mathbf{y}}\mathbf{B}_{ub}\mathbf{V}= \frac{1}{Red\mathbf{x}^2}(\mathbf{D}_1\mathbf{U}+\mathbf{b}_{1u})+\frac{1}{Red\mathbf{y}^2}(\mathbf{D}_2\mathbf{U}+\mathbf{b}_{2u}) \tag{A.2a}$$

$$\frac{d\mathbf{V}}{dt}+\mathbf{f}_2(\mathbf{U},\mathbf{V})-\frac{1}{2d\mathbf{x}}\mathbf{B}_{vl}\mathbf{U}-\frac{1}{2d\mathbf{y}}\mathbf{B}_{vb}\mathbf{V}= \frac{1}{Red\mathbf{x}^2}(\mathbf{D}_1\mathbf{V}+\mathbf{b}_{1v})+\frac{1}{Red\mathbf{y}^2}(\mathbf{D}_2\mathbf{V}+\mathbf{b}_{2v}) \tag{A.2b}$$

where

$$\mathbf{U} = (\mathbf{u}_{1,1}, \mathbf{u}_{2,1}, ..., \mathbf{u}_{n,1}, \mathbf{u}_{1,2}, ..., \mathbf{u}_{n,2}, ..., \mathbf{u}_{1,n}, ...\mathbf{u}_{n,n})^T$$
$$\mathbf{V} = (\mathbf{v}_{1,1}, \mathbf{v}_{2,1}, ..., \mathbf{v}_{n,1}, \mathbf{v}_{1,2}, ..., \mathbf{v}_{n,2}, ..., \mathbf{v}_{1,n}, ...\mathbf{v}_{n,n})^T$$

and the nonlinear mappings $\mathbf{f}_1(\mathbf{U},\mathbf{V})$ and $\mathbf{f}_2(\mathbf{U},\mathbf{V})$ are defined as:

$$\mathbf{f}_1(\mathbf{U},\mathbf{V}) = \frac{1}{2d\mathbf{x}}\mathbf{M}\mathbf{U}.*\mathbf{U}+\frac{1}{2d\mathbf{y}}\mathbf{N}\mathbf{U}.*\mathbf{V}, \tag{A.3a}$$

$$\mathbf{f}_2(\mathbf{U},\mathbf{V}) = \frac{1}{2d\mathbf{x}}\mathbf{M}\mathbf{V}.*\mathbf{U}+\frac{1}{2d\mathbf{y}}\mathbf{N}\mathbf{V}.*\mathbf{V} \tag{A.3b}$$

and

$$\mathbf{M} = \begin{pmatrix} \mathbf{M}_1 & & \\ & \ddots & \\ & & \mathbf{M}_1 \end{pmatrix}_{(n_y-2)\times(n_y-2)}, \mathbf{M}_1 = \begin{pmatrix} 0 & 1 & \\ -1 & \ddots & 1 \\ & -1 & 0 \end{pmatrix}_{(n_x-2)\times(n_x-2)}$$

Also,

$$\begin{aligned}
\mathbf{B}_{ul} &= diag(kron(\mathbf{u}(1, 2 : n_y - 1), [1, 0, 0, ..., 0]_{1\times(n_x-2)})),\\
\mathbf{B}_{ub} &= diag(kron([1, 0, 0, ..., 0]_{1\times(n_y-2)}, \mathbf{u}(2 : n_x - 1), 1)),\\
\mathbf{b}_{1u} &= (\mathbf{u}_{1,2}, 0, ..., 0, \mathbf{u}_{n_{y,2},\mathbf{u}_{1,3}}, 0, ...\mathbf{u}_{n_{x-1},n_y})^T\\
\mathbf{b}_{2u} &= [\mathbf{u}(1, 2 : n_y - 1)^T, 0, ..., 0, \mathbf{u}(n_x, 2 : n_y - 1)^T]^T
\end{aligned}$$

are used to define the boundary conditions where:

$$\mathbf{D}_1 = \begin{pmatrix} \mathbf{D}_{11} & & \\ & \ddots & \\ & & \mathbf{D}_{11} \end{pmatrix}_{(n_y-2)\times(n_y-2)}, \mathbf{D}_2 = \begin{pmatrix} -2\mathbf{E} & \mathbf{E} & \\ \mathbf{E} & \ddots & \mathbf{E} \\ & \mathbf{E} & -2\mathbf{E} \end{pmatrix}_{(n_y-2)\times(n_y-2)}$$

$$\mathbf{D}_{11} = \begin{pmatrix} -2 & 1 & \\ 1 & \ddots & 1 \\ & 1 & -2 \end{pmatrix}_{(n_x-2)\times(n_x-2)}, \quad \mathbf{N} = \begin{pmatrix} \mathbf{E} & & \\ -\mathbf{E} & \ddots & \\ & -\mathbf{E} & \mathbf{E} \end{pmatrix}$$

where $\mathbf{E}$ is $(n_x - 2) \times (n_y - 2)$ identity matrix. Similarly, $\mathbf{b}_{1v}, \mathbf{b}_{2v}, \mathbf{B}_{vl}$ and $\mathbf{B}_{vb}$ can be defined for $\mathbf{v}_{i,j}$. The initial conditions and boundary conditions for this case are taken from Ref. [115] given as.

$$\mathbf{u}(\mathbf{x}, \mathbf{y}, t) = \frac{3}{4} - \frac{1}{4[1 + exp((-4\mathbf{x} + 4\mathbf{y} - t)Re/32)]} \tag{A.4a}$$

$$\mathbf{v}(\mathbf{x}, \mathbf{y}, t) = \frac{3}{4} + \frac{1}{4[1 + exp((-4\mathbf{x} + 4\mathbf{y} - t)Re/32)]} \tag{A.4b}$$

Bibliography

[1] Acle, Y. G. I., Freitas, F. D., Martins, N., and Rommes, J. Parameter preserving model order reduction of large sparse small-signal electromechanical stability power system models. *IEEE Transactions on Power Systems 34*, 4 (2019), 2814–2824.

[2] Al-Baiyat, S. A., and Bettayeb, M. A new model reduction scheme for k-power bilinear systems. In *Proceedings of 32nd IEEE Conference on Decision and Control* (1993), IEEE, pp. 22–27.

[3] Al-Baiyat, S. A., Beyttayeb, M., and Al-Saggaf, U. M. New model reduction scheme for bilinear systems. *International Journal of Systems Science 25*, 10 (1994), 1631–1642.

[4] Aliaga, J., Boley, D., Freund, R., and Hernández, V. A lanczos-type method for multiple starting vectors. *Mathematics of computation 69*, 232 (2000), 1577–1601.

[5] Alla, A., and Kutz, J. N. Nonlinear model order reduction via dynamic mode decomposition. *SIAM Journal on Scientific Computing 39*, 5 (2017), B778–B796.

[6] Amsallem, D., Cortial, J., Carlberg, K., and Farhat, C. A method for interpolating on manifolds structural dynamics reduced-order models. *International journal for numerical methods in engineering 80*, 9 (2009), 1241–1258.

[7] Amsallem, D., Deolalikar, S., Gurrola, F., and Farhat, C. Model predictive control under coupled fluid-structure constraints using a database of reduced-order models on a tablet. In *21st AIAA Computational Fluid Dynamics Conference* (2013), p. 2588.

[8] Amsallem, D., and Farhat, C. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA Journal 46*, 7 (2008), 1803–1813.

[9] Amsallem, D., and Farhat, C. An online method for interpolating linear parametric reduced-order models. *SIAM Journal on Scientific Computing 33*, 5 (2011), 2169–2198.

[10] Amsallem, D., Zahr, M. J., and Farhat, C. Nonlinear model order reduction based on local reduced-order bases. *International Journal for Numerical Methods in Engineering 92*, 10 (2012), 891–916.

[11] Anić, B., Beattie, C., Gugercin, S., and Antoulas, A. C. Interpolatory weighted-$\mathcal{H}_2$ model reduction. *Automatica 49*, 5 (2013), 1275–1280.

[12] Antil, H., Heinkenschloss, M., and Hoppe, R. H. Domain decomposition and balanced truncation model reduction for shape optimization of the stokes system. *Optimization Methods and Software 26*, 4-5 (2011), 643–669.

[13] Antil, H., Heinkenschloss, M., Hoppe, R. H., Linsenmann, C., and Wixforth, A. Reduced order modeling based shape optimization of surface acoustic wave driven microfluidic biochips. *Mathematics and Computers in Simulation 82*, 10 (2012), 1986–2003.

[14] Antoulas, A. C. *Approximation of large-scale dynamical systems.* SIAM,Philadelphia, 2005.

[15] Antoulas, A. C. A new result on passivity preserving model reduction. *Systems & control letters 54*, 4 (2005), 361–374.

[16] Antoulas, A. C. An overview of approximation methods for large-scale dynamical systems. *Annual reviews in Control 29*, 2 (2005), 181–190.

[17] Antoulas, A. C., Beattie, C. A., and Gugercin, S. Interpolatory model reduction of large-scale dynamical systems. In *Efficient modeling and control of large-scale systems*. Springer, 2010, pp. 3–58.

[18] Antoulas, A. C., Beattie, C. A., and Gugercin, S. *Interpolatory methods for model reduction.* SIAM,Philadelphia, 2020.

[19] Antoulas, A. C., Sorensen, D. C., and Gugercin, S. A survey of model reduction methods for large-scale systems. *ContempMath 280* (2001), 193–219.

[20] Aridhi, H., Zaki, M. H., and Tahar, S. Towards improving simulation of analog circuits using model order reduction. In *Design, Automation Test in Europe Conference Exhibition* (2012), pp. 1337–1342.

[21] Ariff, M., and Pal, B. C. Coherency identification in interconnected power system—an independent component analysis approach. *IEEE Transactions on Power Systems 28*, 2 (2012), 1747–1755.

[22] Arnoldi, W. E. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of applied mathematics 9*, 1 (1951), 17–29.

[23] Astolfi, A. Model reduction by moment matching. In *7th IFAC Symposium on Nonlinear Control Systems* (2007), Elsevier, pp. 577–584.

[24] Astolfi, A. A new look at model reduction by moment matching for linear systems. In *2007 46th IEEE Conference on Decision and Control* (2007), IEEE, pp. 4361–4366.

[25] Astolfi, A. Model reduction by moment matching for nonlinear systems. In *2008 47th IEEE Conference on Decision and Control* (2008), IEEE, pp. 4873–4878.

[26] Astolfi, A. Model reduction by moment matching for linear and nonlinear systems. *IEEE Transactions on Automatic Control 55*, 10 (2010), 2321–2336.

[27] Astolfi, A. Model reduction by moment matching, steady-state response and projections. In *49th IEEE Conference on Decision and Control (CDC)* (2010), IEEE, pp. 5344–5349.

[28] Astrid, P., Weiland, S., Willcox, K., and Backx, T. Missing point estimation in models described by proper orthogonal decomposition. *IEEE Transactions on Automatic Control 53*, 10 (2008), 2237–2251.

[29] Åström, K. J., and Wittenmark, B. *Adaptive control*. Addison-Wesley Longman Publising Compant, 2nd edition, 1994.

[30] Bahadir, A. R. A fully implicit finite-difference scheme for two-dimensional burgers' equations. *Applied Mathematics and Computation 137*, 1 (2003), 131–137.

[31] Bai, Z. Krylov subspace techniques for reduced-order modeling of large-scale dynamical systems. *Applied numerical mathematics 43*, 1-2 (2002), 9–44.

[32] Bai, Z., Feldmann, P., and Freund, R. W. Stable and passive reduced-order models based on partial padé approximation via the lanczos process. *Numerical Analysis Manuscript 97*, 3 (1997), 10.

[33] Bai, Z., and Skoogh, D. A projection method for model reduction of bilinear dynamical systems. *Linear algebra and its applications 415*, 2-3 (2006), 406–425.

[34] Bai, Z., Slone, R. D., Smith, W. T., and Ye, Q. Error bound for reduced system model by padé approximation via the lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 18*, 2 (1999), 133–141.

[35] Bai, Z., and Su, Y. Dimension reduction of large-scale second-order dynamical systems via a second-order arnoldi method. *SIAM Journal on Scientific Computing 26*, 5 (2005), 1692–1709.

[36] Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T. An empirical interpolation method: application to efficient reduced-basis discretization of partial differential equations. *Comptes Rendus Mathematique 339*, 9 (2004), 667–672.

[37] Bastian, J., Haase, J., et al. Order reduction of second order systems. In *In Proc. 4th Mathmod* (2003), Citeseer.

[38] Baur, U., and Benner, P. Gramian-based model reduction for data-sparse systems. *SIAM Journal on Scientific Computing 31*, 1 (2008), 776–798.

[39] Baur, U., and Benner, P. Model reduction for parametric systems using balanced truncation and interpolation. *at-Automatisierungstechnik 57*, 8 (2009), 411–419.

[40] Baur, U., Benner, P., and Feng, L. Model order reduction for linear and nonlinear systems: a system-theoretic perspective. *Archives of Computational Methods in Engineering 21*, 4 (2014), 331–358.

[41] Beattie, C., and Gugercin, S. Realization-independent $\mathcal{H}_2$-approximation. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)* (2012), IEEE, pp. 4953–4958.

[42] Beattie, C. A., and Gugercin, S. Krylov-based minimization for optimal h 2 model reduction. In *2007 46th IEEE Conference on Decision and Control* (2007), IEEE, pp. 4385–4390.

[43] Beattie, C. A., and Gugercin, S. A trust region method for optimal $\mathcal{H}_2$ model reduction. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference* (2009), IEEE, pp. 5370–5375.

[44] Bechtold, T., Rudnyi, E. B., and Korvink, J. G. Error indicators for fully automatic extraction of heat-transfer macromodels for mems. *Journal of Micromechanics and Microengineering 15*, 3 (2004), 430.

[45] Benner, P. Solving large-scale control problems. *IEEE Control Systems Magazine 24*, 1 (2004), 44–59.

[46] Benner, P., and Breiten, T. Interpolation-based $\mathcal{H}_2$model reduction of bilinear control systems. *SIAM Journal on Matrix Analysis and Applications 33*, 3 (2012), 859–885.

[47] Benner, P., and Breiten, T. Krylov-subspace based model reduction of nonlinear circuit models using bilinear and quadratic-linear approximations. In *Progress in Industrial Mathematics at ECMI 2010*. Springer, 2012, pp. 153–159.

[48] Benner, P., and Breiten, T. Two-sided projection methods for nonlinear model order reduction. *SIAM Journal on Scientific Computing 37*, 2 (2015), B239–B260.

[49] Benner, P., and Damm, T. Lyapunov equations, energy functionals and model order reduction. *Preprint, TU Chemnitz* (2009).

[50] Benner, P., and Damm, T. Lyapunov equations, energy functionals, and model order reduction of bilinear and stochastic systems. *SIAM journal on control and optimization 49*, 2 (2011), 686–711.

[51] Benner, P., and Feng, L. A robust algorithm for parametric model order reduction based on implicit moment matching. In *Reduced Order Methods for Modeling and Computational Reduction*. Springer, 2014, pp. 159–185.

[52] Benner, P., Feng, L., Li, S., and Zhang, Y. Reduced-order modeling and ROM-based optimization of batch chromatography. In *Numerical Mathematics and Advanced Applications-ENUMATH 2013*. Springer, 2015, pp. 427–435.

[53] Benner, P., Gugercin, S., and Willcox, K. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM review 57*, 4 (2015), 483–531.

[54] Benner, P., Hinze, M., and Ter Maten, E. J. W. *Model reduction for circuit simulation*, vol. 74. Springer, 2011.

[55] Benner, P., Køhler, M., and Saak, J. Sparse-dense sylvester equations in $\mathcal{H}_2$-model order reduction. *Technical Report MPIMD* (2011).

[56] Benner, P., and Stykel, T. Model order reduction for differential-algebraic equations: a survey. In *Surveys in Differential-Algebraic Equations IV*. Springer, 2017, pp. 107–160.

[57] Benoît, E. *Dynamic bifurcations: proceedings of a conference held in Luminy, France, March 5-10, 1990*. Springer, 2006.

[58] Berkooz, G., Holmes, P., and Lumley, J. L. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics 25*, 1 (1993), 539–575.

[59] Boley, D. L. Krylov space methods on state-space control models. *Circuits, Systems and Signal Processing 13*, 6 (1994), 733–758.

[60] Bollhöfer, M., and Bodendiek, A. Adaptive-order rational arnoldi method for maxwell's equations. *Scientific computing in electrical engineering (Abstracts)* (2012), 77–78.

[61] Bond, B., and Daniel, L. Parameterized model order reduction of nonlinear dynamical systems. In *ICCAD-2005. IEEE/ACM International Conference on Computer-Aided Design, 2005.* (2005), IEEE, pp. 487–494.

[62] Breiten, T. *Interpolatory methods for model reduction of large-scale dynamical systems*. PhD thesis, Otto-von-Guericke Universität Magdeburg, 2013.

[63] Breiten, T., Beattie, C., and Gugercin, S. Near-optimal frequency-weighted interpolatory model reduction. *Systems & Control Letters 78* (2015), 8–18.

[64] Brunton, S. L., and Kutz, J. N. *Data-driven science and engineering: Machine learning, dynamical systems, and control*. Cambridge University Press, 2019.

[65] Bui-Thanh, T., Willcox, K., and Ghattas, O. Parametric reduced-order models for probabilistic analysis of unsteady aerodynamic applications. *AIAA Journal 46*, 10 (2008), 2520–2529.

[66] Bunse-Gerstner, A., Kubalińska, D., Vossen, G., and Wilczek, D. $\mathcal{H}_2$-norm optimal model reduction for large scale discrete dynamical MIMO systems. *Journal of computational and applied mathematics 233*, 5 (2010), 1202–1216.

[67] Byrnes, C., and Isidori, A. Steady state response, separation principle and the output regulation of nonlinear systems. In *Proceedings of the 28th IEEE Conference on Decision and Control,* (1989), IEEE, pp. 2247–2251.

[68] Byrnes, C. I., Lindquist, A., Gusev, S. V., and Matveev, A. S. A complete parameterization of all positive rational extensions of a covariance sequence. *IEEE Transactions on Automatic Control 40*, 11 (1995), 1841–1857.

[69] Carlberg, K., Bou-Mosleh, C., and Farhat, C. Efficient non-linear model reduction via a least-squares petrov–galerkin projection and compressive tensor approximations. *International Journal for Numerical Methods in Engineering 86*, 2 (2011), 155–181.

[70] Carlberg, K., Farhat, C., Cortial, J., and Amsallem, D. The gnat method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. *Journal of Computational Physics 242* (2013), 623–647.

[71] Carlberg, K., Tuminaro, R., and Boggs, P. Preserving lagrangian structure in nonlinear model reduction with application to structural dynamics. *SIAM Journal on Scientific Computing 37*, 2 (2015), B153–B184.

[72] Carr, J. *Applications of Center Manifold Theory.* volume 35, Applied Mathematical Sciences, 1982.

[73] Cepeda, J. C., Rueda, J. L., and Erlich, I. Identification of dynamic equivalents based on heuristic optimization for smart grid applications. In *2012 IEEE Congress on Evolutionary Computation* (2012), IEEE, pp. 1–8.

[74] Chafee, N., and Infante, E. Bifurcation and stability for a nonlinear parabolic partial differential equation. *Bulletin of the American Mathematical Society 80*, 1 (1974), 49–52.

[75] Chahlaoui, Y., Lemonnier, D., Vandendorpe, A., and Van Dooren, P. Second order structure preserving balanced truncation. *Symp. on Math. Theory of Network and Systems* (2004).

[76] Chahlaoui, Y., Lemonnier, D., Vandendorpe, A., and Van Dooren, P. Second-order balanced truncation. *Linear algebra and its applications 415*, 2-3 (2006), 373–384.

[77] Chakrabortty, A., Chow, J. H., and Salazar, A. A measurement-based framework for dynamic equivalencing of large power systems using wide-area phasor measurements. *IEEE Transactions on Smart Grid 2*, 1 (2010), 68–81.

[78] Chaniotis, D., and Pai, M. Model reduction in power systems using krylov subspace methods. *IEEE Transactions on Power Systems 20*, 2 (2005), 888–894.

[79] Chaturantabut, S., Beattie, C., and Gugercin, S. Structure-preserving model reduction for nonlinear port-hamiltonian systems. *SIAM Journal on Scientific Computing 38*, 5 (2016), B837–B865.

[80] Chaturantabut, S., and Sorensen, D. C. Nonlinear model reduction via discrete empirical interpolation. *SIAM Journal on Scientific Computing 32*, 5 (2010), 2737–2764.

[81] Chen, P. *Model order reduction techniques for uncertainty quantification problems.* PhD thesis, ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE, 2014.

[82] Chen, Y. *Model order reduction for nonlinear systems.* PhD thesis, Massachusetts Institute of Technology, 1999.

[83] Chen, Y., Balakrishnan, V., Koh, C.-K., and Roy, K. Model reduction in the time-domain using laguerre polynomials and krylov methods. In *Proceedings 2002 Design, Automation and Test in Europe Conference and Exhibition* (2002), IEEE, pp. 931–935.

[84] Chiprout, E., and Nakhla, M. S. Asymptotic waveform evaluation. In *Asymptotic Waveform Evaluation*. Springer, 1994, pp. 15–39.

[85] Choi, Y., and Carlberg, K. Space–time least-squares petrov–galerkin projection for nonlinear model reduction. *SIAM Journal on Scientific Computing 41*, 1 (2019), A26–A58.

[86] Chow, J. H. *Power system coherency and model reduction*, vol. 84. Springer, 2013.

[87] Constantin, P., Foias, C., Nicolaenko, B., and Temam, R. *Integral manifolds and inertial manifolds for dissipative partial differential equations*, vol. 70. Springer Science & Business Media, 2012.

[88] Cruz Varona, M., Pak, M., and Lohmann, B. Towards output krylov subspace-based nonlinear moment matching. In *Joint 8th IFAC Symposium on Mechatronic Systems and 11th IFAC Symposium on Nonlinear Control Systems, Vienna, Austria* (2019).

[89] Cullum, J. K., and Willoughby, R. A. *Lanczos algorithms for large symmetric eigenvalue computations: Vol. I: Theory*. SIAM, 2002.

[90] Daneshpajooh, H., and Zand, M. M. Semi-analytic solutions to oscillatory behavior of initially curved micro/nano systems. *Journal of Mechanical Science and Technology 29*, 9 (2015), 3855–3863.

[91] Degroote, J., Vierendeels, J., and Willcox, K. Interpolation among reduced-order matrices to obtain parameterized models for design, optimization and probabilistic analysis. *International Journal for Numerical Methods in Fluids 63*, 2 (2010), 207–230.

[92] Desai, U., and Pal, D. A transformation approach to stochastic model reduction. *IEEE Transactions on Automatic Control 29*, 12 (1984), 1097–1100.

[93] Dong, N., and Roychowdhury, J. Piecewise polynomial nonlinear model reduction. In *Proceedings 2003. Design Automation Conference (IEEE Cat. No. 03CH37451)* (2003), IEEE, pp. 484–489.

[94] Drmac, Z., and Gugercin, S. A new selection operator for the discrete empirical interpolation method—improved a priori error bound and extensions. *SIAM Journal on Scientific Computing 38*, 2 (2016), A631–A648.

[95] Drohmann, M., Haasdonk, B., and Ohlberger, M. Adaptive reduced basis methods for nonlinear convection–diffusion equations. In *Finite Volumes for Complex Applications VI Problems & Perspectives*. Springer, 2011, pp. 369–377.

[96] Druskin, V., and Simoncini, V. Adaptive rational krylov subspaces for large-scale dynamical systems. *Systems & Control Letters 60*, 8 (2011), 546–560.

[97] Druskin, V., Simoncini, V., and Zaslavsky, M. Solution of the time-domain inverse resistivity problem in the model reduction framework part i. one-dimensional problem with siso data. *SIAM Journal on Scientific Computing 35*, 3 (2013), A1621–A1640.

[98] Eckart, C., and Young, G. The approximation of one matrix by another of lower rank. *Psychometrika 1*, 3 (1936), 211–218.

[99] Eftang, J. L., and Stamm, B. Parameter multi-domain 'hp'empirical interpolation. *International Journal for Numerical Methods in Engineering 90*, 4 (2012), 412–428.

[100] Eid, R. *Time domain model reduction by moment matching*. PhD thesis, Technische Universität München, 2009.

[101] Elman, H. C., and Liao, Q. Reduced basis collocation methods for partial differential equations with random coefficients. *SIAM/ASA Journal on Uncertainty Quantification 1*, 1 (2013), 192–217.

[102] Enns, D. F. Model reduction with balanced realizations: An error bound and a frequency weighted generalization. In *The 23rd IEEE Conference on Decision and Control* (1984), IEEE, pp. 127–132.

[103] Everson, R., and Sirovich, L. Karhunen–loeve procedure for gappy data. *JOSA A 12*, 8 (1995), 1657–1664.

[104] Faedo, N., Piuma, F. J. D., Giorgi, G., and Ringwood, J. V. Nonlinear model reduction for wave energy systems: a moment-matching-based approach. *Nonlinear Dynamics 102*, 3 (2020), 1215–1237.

[105] Fahl, M., and Sachs, E. W. Reduced order modelling approaches to pde-constrained optimization based on proper orthogonal decomposition. In *Large-scale PDE-constrained optimization*. Springer, 2003, pp. 268–280.

[106] Farhat, C., Avery, P., Chapman, T., and Cortial, J. Dimensional reduction of nonlinear finite element dynamic models with finite rotations and energy-based mesh sampling and weighting for computational efficiency. *International Journal for Numerical Methods in Engineering 98*, 9 (2014), 625–662.

[107] Feldmann, P., and Freund, R. W. Efficient linear circuit analysis by padé approximation via the lanczos process. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 14*, 5 (1995), 639–649.

[108] Feng, L., and Benner, P. Automatic model order reduction by moment-matching according to an efficient output error bound. *Scientific computing in electrical engineering (Abstracts)* (2012), 71–72.

[109] Feng, L., Benner, P., and Korvink, J. G. System-level modeling of mems by means of model order reduction (mathematical approximations)–mathematical background. *System-Level Modeling of MEMS* (2013), 53–93.

[110] Feng, L., Korvink, J. G., and Benner, P. A fully adaptive scheme for model order reduction based on moment matching. *IEEE Transactions on Components, Packaging and Manufacturing Technology 5*, 12 (2015), 1872–1884.

[111] Feng, L., Koziol, D., Rudnyi, E. B., and Korvink, J. G. Parametric model reduction for fast simulation of cyclic voltammograms. *Sensor Letters 4*, 2 (2006), 165–173.

[112] Feng, L., Zeng, X., Chiang, C., Zhou, D., and Fang, Q. Direct nonlinear order reduction with variational analysis. In *Proceedings Design, Automation and Test in Europe Conference and Exhibition* (2004), vol. 2, IEEE, pp. 1316–1321.

[113] Feng, L. H., Rudnyi, E. B., and Korvink, J. G. Preserving the film coefficient as a parameter in the compact thermal model for fast electrothermal simulation. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 24*, 12 (2005), 1838–1847.

[114] Flagg, G., Beattie, C., and Gugercin, S. Convergence of the iterative rational krylov algorithm. *Systems & Control Letters 61*, 6 (2012), 688–691.

[115] Fletcher, C. A. Generating exact solutions of the two-dimensional burgers' equations. *International Journal for Numerical Methods in Fluids 3* (1983), 213–216.

[116] Freund, R. W. Reduced-order modeling techniques based on krylov subspaces and their use in circuit simulation. In *Applied and computational control, signals, and circuits*. Springer, 1999, pp. 435–498.

[117] Freund, R. W. Krylov-subspace methods for reduced-order modeling in circuit simulation. *Journal of Computational and Applied Mathematics 123*, 1-2 (2000), 395–421.

[118] Freund, R. W. Passive reduced-order modeling via krylov-subspace methods. In *CACSD. Conference Proceedings. IEEE International Symposium on Computer-Aided Control System Design (Cat. No. 00TH8537)* (2000), IEEE, pp. 261–266.

[119] Freund, R. W. Model reduction methods based on krylov subspaces. *Acta Numerica 12* (2003), 267–319.

[120] Freund, R. W. SPRIM: structure-preserving reduced-order interconnect macromodeling. In *IEEE/ACM International Conference on Computer Aided Design, 2004. ICCAD-2004.* (2004), IEEE, pp. 80–87.

[121] Fujimoto, K. Balanced realization and model order reduction for port-hamiltonian systems. *Journal of System Design and Dynamics 2*, 3 (2008), 694–702.

[122] Fujimoto, K., and Scherpen, J. M. Nonlinear input-normal realizations based on the differential eigenstructure of hankel operators. *IEEE Transactions on Automatic Control 50*, 1 (2005), 2–18.

[123] Fujimoto, K., and Scherpen, J. M. Balanced realization and model order reduction for nonlinear systems based on singular value analysis. *SIAM Journal on Control and Optimization 48*, 7 (2010), 4591–4623.

[124] Gallivan, K., Grimme, E., and Dooren, P. V. Asymptotic waveform evaluation via a lanczos method. *Applied Mathematics Letters 7*, 5 (1994), 75–80.

[125] Gallivan, K., Vandendorpe, A., and Van Dooren, P. Model reduction of MIMO systems via tangential interpolation. *SIAM Journal on Matrix Analysis and Applications 26*, 2 (2004), 328–349.

[126] Gallivan, K., Vandendorpe, A., and Van Dooren, P. Sylvester equations and projection-based model reduction. *Journal of Computational and Applied Mathematics 162*, 1 (2004), 213–229.

[127] Gallivan, K., Vandendorpe, A., and Van Dooren, P. Model reduction and the solution of sylvester equations. *MTNS, Kyoto 50* (2006).

[128] Germond, A. J., and Podmore, R. Dynamic aggregation of generating unit models. *IEEE Transactions on Power Apparatus and Systems*, 97 (1978), 1060–1069.

[129] Geuss, M., Panzer, H., and Lohmann, B. On parametric model order reduction by matrix interpolation. In *2013 European Control Conference (ECC)* (2013), IEEE, pp. 3433–3438.

[130] Glover, K. All optimal hankel-norm approximations of linear multivariable systems and their $\mathcal{L}_\infty$-error bounds. *International Journal of Control 39*, 6 (1984), 1115–1193.

[131] Goyal, P. K. *System-theoretic model order reduction for bilinear and quadratic-bilinear systems*. PhD thesis, Universitätsbibliothek, 2018.

[132] Gragg, W. B., and Lindquist, A. On the partial realization problem. *Linear Algebra and its Applications 50* (1983), 277–319.

[133] Gray, W. S., and Mesko, J. General input balancing and model reduction for linear and nonlinear systems. In *1997 European Control Conference (ECC)* (1997), IEEE, pp. 2862–2867.

[134] Grepl, M. A. *Reduced-basis approximation a posteriori error estimation for parabolic partial differential equations*. PhD thesis, Massachusetts Institute of Technology, 2005.

[135] Grepl, M. A., Maday, Y., Nguyen, N. C., and Patera, A. T. Efficient reduced-basis treatment of nonaffine and nonlinear partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis 41*, 3 (2007), 575–605.

[136] Grepl, M. A., and Patera, A. T. A posteriori error bounds for reduced-basis approximations of parametrized parabolic partial differential equations. *ESAIM: Mathematical Modelling and Numerical Analysis 39*, 1 (2005), 157–181.

[137] Grimme, E. *Krylov projection methods for model reduction*. PhD thesis, University of Illinois at Urbana Champaign, 1997.

[138] Grimme, E. J., Sorensen, D. C., and Van Dooren, P. Model reduction of state space systems via an implicitly restarted lanczos method. *Numerical algorithms 12*, 1 (1996), 1–31.

[139] Gu, C. QLMOR: A projection-based nonlinear model order reduction approach using quadratic-linear representation of nonlinear systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 30*, 9 (2011), 1307–1320.

[140] Gugercin, S. An iterative rational krylov algorithm (IRKA) for optimal $\mathcal{H}_2$ model reduction. In *Householder Symposium XVI, Seven Springs Mountain Resort, PA, USA* (2005).

[141] Gugercin, S., Antoulas, A. C., and Beattie, C. $\mathcal{H}_2$ model reduction for large-scale linear dynamical systems. *SIAM journal on matrix analysis and applications 30*, 2 (2008), 609–638.

[142] Gugercin, S., Beattie, C., and Antoulas, A. Rational krylov methods for optimal $\mathcal{H}_2$ model reduction. *submitted for publication* (2006).

[143] Gugercin, S., and Li, J.-R. Smith-type methods for balanced truncation of large sparse systems. In *Dimension reduction of large-scale systems*. Springer, 2005, pp. 49–82.

[144] Gugercin, S., Sorensen, D. C., and Antoulas, A. C. A modified low-rank smith method for large-scale lyapunov equations. *Numerical Algorithms 32*, 1 (2003), 27–55.

[145] Gugercin S, S. T., and S, W. Model reduction of descriptor systems by interpolatory projection methods. *SIAM Journal on Scientific Computing 35* (2013), 1010–1033.

[146] Gunupudi, P. K., and Nakhla, M. S. Model-reduction of nonlinear circuits using krylov-space techniques. In *Proceedings of the 36th annual ACM/IEEE Design Automation Conference* (1999), pp. 13–16.

[147] Haasdonk, B., Dihlmann, M., and Ohlberger, M. A training set and multiple bases generation approach for parameterized model reduction based on adaptive grids in parameter space. *Mathematical and Computer Modelling of Dynamical Systems 17*, 4 (2011), 423–442.

[148] Halevi, Y. Frequency weighted model reduction via optimal projection. In *29th IEEE Conference on Decision and Control* (1990), IEEE, pp. 2906–2911.

[149] Hesthaven, J. S., Rozza, G., Stamm, B., et al. *Certified reduced basis methods for parametrized partial differential equations*, vol. 590. Springer, 2016.

[150] Hinze, M., and Volkwein, S. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control. In *Dimension reduction of large-scale systems*. Springer, 2005, pp. 261–306.

[151] Hochman, A., Vasilyev, D. M., Rewienski, M. J., and White, J. K. Projection-based nonlinear model order reduction. *System-level modeling of MEMS, advanced micro & nanosystems. Wiley-VCH* (2013).

[152] Horn, R. A., and Johnson, C. R. *Matrix analysis*. Cambridge University Press, 2012.

[153] Hovland, S., Gravdahl, J. T., and Willcox, K. E. Explicit model predictive control for large-scale systems via model reduction. *Journal of Guidance, Control, and Dynamics 31*, 4 (2008), 918–926.

[154] Ionescu, T. C., and Astolfi, A. Families of reduced order models that achieve nonlinear moment matching. In *2013 American Control Conference* (2013), IEEE, pp. 5518–5523.

[155] Ionescu, T. C., and Astolfi, A. Nonlinear moment matching-based model order reduction. *IEEE Transactions on Automatic Control 61*, 10 (2015), 2837–2847.

[156] Ionescu, T. C., Astolfi, A., and Colaneri, P. Families of moment matching based, low order approximations for linear systems. *Systems & Control Letters 64* (2014), 47–56.

[157] Isidori, A. *Nonlinear Control Systems*. Springer Science & Business Media, Third Edition, 1995.

[158] Isidori, A., and Byrnes, C. I. Output regulation of nonlinear systems. *IEEE transactions on Automatic Control 35*, 2 (1990), 131–140.

[159] Jaimoukha, I. M., and Kasenally, E. M. Oblique production methods for large scale model reduction. *SIAM Journal on Matrix Analysis and Applications 16*, 2 (1995), 602–627.

[160] Jaimoukha, I. M., and Kasenally, E. M. Implicitly restarted krylov subspace methods for stable partial realizations. *SIAM Journal on Matrix Analysis and Applications 18*, 3 (1997), 633–652.

[161] Jonckheere, E., and Silverman, L. A new of invariants for linear system-application to reduced order compensator design. *IEEE Transactions on Automatic Control 28* (1983), 953–964.

[162] Jordan, D., Smith, P., and Smith, P. *Nonlinear ordinary differential equations: an introduction for scientists and engineers*, vol. 10. Oxford University Press on Demand, 2007.

[163] Kaczynski, J., Ranacher, C., and Fleury, C. Computationally efficient model for viscous damping in perforated mems structures. *Sensors and Actuators A: Physical 314* (2020), 112201.

[164] Kawano, Y., and Scherpen, J. M. Model reduction by differential balancing based on nonlinear hankel operators. *IEEE Transactions on Automatic Control 62*, 7 (2016), 3293–3308.

[165] Kellems, A. R., Roos, D., Xiao, N., and Cox, S. J. Low-dimensional, morphologically accurate models of subthreshold membrane potential. *Journal of Computational Neuroscience 27*, 2 (2009), 161.

[166] Kerns, K. J., and Yang, A. T. Preservation of passivity during rlc network reduction via split congruence transformations. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 17*, 7 (1998), 582–591.

[167] Kim, H. M., and Craig Jr, R. R. Structural dynamics analysis using an unsymmetric block lanczos algorithm. *International journal for numerical methods in engineering 26*, 10 (1988), 2305–2318.

[168] Kim, H. M., and Craig Jr, R. R. Computational enhancement of an unsymmetric block lanczos algorithm. *International journal for numerical methods in engineering 30*, 5 (1990), 1083–1089.

[169] Konkel, Y., Farle, O., Sommer, A., Burgard, S., and Dyczij-Edlinger, R. A posteriori error bounds for krylov-based fast frequency sweeps of finite-element systems. *IEEE Transactions on Magnetics 50*, 2 (2014), 441–444.

[170] Koopman, B. O. Hamiltonian systems and transformation in hilbert space. *Proceedings of the national academy of sciences of the united states of america 17*, 5 (1931), 315.

[171] Krack, M., and Gross, J. *Harmonic balance for nonlinear vibration problems*. Springer, 2019.

[172] Krajewski, W., Lepschy, A., Redivo-Zaglia, M., and Viaro, U. A program for solving the l 2 reduced-order model problem with fixed denominator degree. *Numerical Algorithms 9*, 2 (1995), 355–377.

[173] Krener, A. J. The construction of optimal linear and nonlinear regulators. In *Systems, Models and Feedback: Theory and Applications*. Springer, 1992, pp. 301–322.

[174] Kunisch, K., and Volkwein, S. Control of the burgers equation by a reduced-order approach using proper orthogonal decomposition. *Journal of optimization theory and applications 102*, 2 (1999), 345–371.

[175] Kunisch, K., and Volkwein, S. Proper orthogonal decomposition for optimality systems. *ESAIM: Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique 42*, 1 (2008), 1–23.

[176] Kutz, J. N., Brunton, S. L., Brunton, B. W., and Proctor, J. L. *Dynamic mode decomposition: data-driven modeling of complex systems*. SIAM, 2016.

[177] Lall, S., Krysl, P., and Marsden, J. E. Structure-preserving model reduction for mechanical systems. *Physica D: Nonlinear Phenomena 184*, 1-4 (2003), 304–318.

[178] Lall, S., Marsden, J. E., and Glavaški, S. Empirical model reduction of controlled nonlinear systems. *IFAC Proceedings Volumes 32*, 2 (1999), 2598–2603.

[179] Lall, S., Marsden, J. E., and Glavaški, S. A subspace approach to balanced truncation for model reduction of nonlinear control systems. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal 12*, 6 (2002), 519–535.

[180] Lanczos, C. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*. United States Governm. Press Office Los Angeles, CA, 1950.

[181] Lang, N., Saak, J., and Stykel, T. Balanced truncation model reduction for linear time-varying systems. *Mathematical and Computer Modelling of Dynamical Systems 22*, 4 (2016), 267–281.

[182] Lee, H.-J., Chu, C.-C., and Feng, W.-S. An adaptive-order rational arnoldi method for model-order reductions of linear time-invariant systems. *Linear algebra and its applications 415*, 2-3 (2006), 235–261.

[183] Lee, K., and Carlberg, K. T. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics 404* (2020), 108973.

[184] Lehoucq, R. B., Sorensen, D. C., and Yang, C. *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*. SIAM, 1998.

[185] Li, R.-C., Bai, Z., et al. Structure-preserving model reduction using a krylov subspace projection formulation. *Communications in Mathematical Sciences 3*, 2 (2005), 179–199.

[186] Lieu, T., and Farhat, C. Adaptation of aeroelastic reduced-order models and application to an f-16 configuration. *AIAA Journal 45*, 6 (2007), 1244–1257.

[187] Liu, C.-C., and Wang, C.-C. Numerical investigation into nonlinear dynamic behavior of electrically-actuated clamped–clamped micro-beam with squeeze-film damping effect. *Applied Mathematical Modelling 38*, 13 (2014), 3269–3280.

[188] Ljung, L. *System identification: Theory for the User*. Prentice Hall Information and System Sciences Series, 2nd edition., 1999.

[189] Lohmann, B., and Eid, R. Efficient order reduction of parametric and nonlinear models by superposition of locally reduced models. In *Methoden*

und Anwendungen der Regelungstechnik. Erlangen-Münchener Workshops (2007), pp. 27–36.

[190] Lohmann, B., and Salimbahrami, B. Order reduction using krylov subspace methods. *Automation Technology 52*, 1 (2004), 30–38.

[191] Lohmann, B., and Salimbahrami, B. Reduction of second order systems using second order krylov subspaces. *IFAC Proceedings Volumes 38*, 1 (2005), 614–619.

[192] Lozano, S., Buzna, L., and Díaz-Guilera, A. Role of network topology in the synchronization of power systems. *The European Physical Journal B 85*, 7 (2012), 1–8.

[193] Lumley, J. The structure of inhomogeneous turbulent flows. *Atmospheric Turbulence and Radio Wave Propagation* (1967), 166–178.

[194] Ma, F., and Vittal, V. A hybrid dynamic equivalent using ann-based boundary matching technique. *IEEE Transactions on Power Systems 27*, 3 (2012), 1494–1502.

[195] Majumder, S., McGruer, N., and Adams, G. Adhesion and contact resistance in an electrostatic mems microswitch. In *18th IEEE International Conference on Micro Electro Mechanical Systems, 2005. MEMS 2005.* (2005), IEEE, pp. 215–218.

[196] Malik, M. H., Borzacchiello, D., Chinesta, F., and Diez, P. Reduced order modeling for transient simulation of power systems using trajectory piece-wise linear approximation. *Advanced Modeling and Simulation in Engineering Sciences 3*, 1 (2016), 1–18.

[197] Maloberti, F. *Analog design for CMOS VLSI systems*, vol. 646. Springer Science & Business Media, 2006.

[198] Martins, N., Lima, L. T., and Pinto, H. J. Computing dominant poles of power system transfer functions. *IEEE Transactions on Power Systems 11*, 1 (1996), 162–170.

[199] Martins, N., and Quintão, P. E. Computing dominant poles of power system multivariable transfer functions. *IEEE Transactions on Power Systems 18*, 1 (2003), 152–159.

[200] Martone, R., Formisano, A., Condon, M., and Ivanov, R. Krylov subspaces from bilinear representations of nonlinear systems. *COMPEL-The International Journal For Computation and Mathematics in electrical and electronic engineering* (2007).

[201] Mead, C., and Conway, L. *Introduction to VLSI systems*, vol. 1080. Addison-Wesley Reading, MA, 1980.

[202] Mehrmann, V., and Stykel, T. Balanced truncation model reduction for large-scale systems in descriptor form. In *Dimension Reduction of Large-Scale Systems*. Springer, 2005, pp. 83–115.

[203] Meier, L., and Luenberger, D. Approximation of linear constant systems. *IEEE Transactions on Automatic Control 12*, 5 (1967), 585–588.

[204] Meyer, C. D. *Matrix analysis and applied linear algebra*, vol. 71. SIAM, 2000.

[205] Meyer, D. G. Fractional balanced reduction: Model reduction via fractional representation. *IEEE Transactions on Automatic Control 35*, 12 (1990), 1341–1345.

[206] Meyer, D. G., and Srinivasan, S. Balancing and model reduction for second-order form linear systems. *IEEE Transactions on Automatic Control 41*, 11 (1996), 1632–1644.

[207] Mezić, I. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics 41*, 1 (2005), 309–325.

[208] Mezić, I. Analysis of fluid flows via spectral properties of the koopman operator. *Annual Review of Fluid Mechanics 45* (2013), 357–378.

[209] Mezić, I., and Banaszuk, A. Comparison of systems with complex behavior. *Physica D: Nonlinear Phenomena 197*, 1-2 (2004), 101–133.

[210] Moore, B. Principal component analysis in linear systems: Controllability, observability, and model reduction. *IEEE Transactions on Automatic Control 26*, 1 (1981), 17–32.

[211] Motter, A. E., Myers, S. A., Anghel, M., and Nishikawa, T. Spontaneous synchrony in power-grid networks. *Nature Physics 9*, 3 (2013), 191–197.

[212] Mullis, C., and Roberts, R. Synthesis of minimum roundoff noise fixed point digital filters. *IEEE Transactions on Circuits and Systems 23*, 9 (1976), 551–562.

[213] Nayfeh, A. H. *Perturbation methods*. John Wiley & Sons, 2008.

[214] Nayfeh, A. H., Mook, D. T., and Holmes, P. *Nonlinear oscillations*. Wiley, 1980.

[215] Nayfeh, A. H., Younis, M. I., and Abdel-Rahman, E. M. Reduced-order models for mems applications. *Nonlinear Dynamics 41*, 1 (2005), 211–236.

[216] Necoara, I., and Ionescu, T. C. $\mathcal{H}_2$ model reduction of linear network systems by moment matching and optimization. *IEEE Transactions on Automatic Control 65*, 12 (2020), 5328–5335.

[217] Nguyen, N. C., Patera, A. T., and Peraire, J. A 'best points' interpolation method for efficient approximation of parametrized functions. *International journal for numerical methods in engineering 73*, 4 (2008), 521–543.

[218] Nishikawa, T., and Motter, A. E. Comparative analysis of existing models for power-grid synchronization. *New Journal of Physics 17*, 1 (2015), 015012.

[219] Nour-Omid, B., and Clough, R. W. Dynamic analysis of structures using lanczos co-ordinates. *Earthquake engineering & structural dynamics 12*, 4 (1984), 565–577.

[220] Odabasioglu, A., Celik, M., and Pileggi, L. T. PRIMA: Passive reduced-order interconnect macromodeling algorithm. In *The Best of ICCAD*. Springer, 2003, pp. 433–450.

[221] Opdenacker, P. C., and Jonckheere, E. A. A contraction mapping preserving balanced reduction scheme and its infinity norm error bounds. *IEEE Transactions on Circuits and Systems 35*, 2 (1988), 184–189.

[222] Osipov, D., and Sun, K. Adaptive nonlinear model reduction for fast power system simulation. *IEEE Transactions on Power Systems 33*, 6 (2018), 6746–6754.

[223] P. Benner, M. H., and Termaten, E. Model reduction for circuit simulation. In *Lecture Notes in Electrical Engineering*, vol. 74. Springer-Verlag,Netherland, 2001.

[224] Panzer, H., Mohring, J., Eid, R., and Lohmann, B. Parametric model order reduction by matrix interpolation. *at-Automatisierungstechnik 58*, 8 (2010), 475–484.

[225] Panzer, H. K., Jaensch, S., Wolf, T., and Lohmann, B. A greedy rational krylov method for $\mathcal{H}_2$ pseudooptimal model order reduction with preservation of stability. In *2013 American Control Conference* (2013), IEEE, pp. 5512–5517.

[226] Panzer, H. K., Wolf, T., and Lohmann, B. $\mathcal{H}_2$ and $\mathcal{H}_\infty$ error bounds for model order reduction of second order systems by krylov subspace methods. In *2013 European Control Conference (ECC)* (2013), IEEE, pp. 4484–4489.

[227] Pawar, S., Rahman, S., Vaddireddy, H., San, O., Rasheed, A., and Vedula, P. A deep learning enabler for nonintrusive reduced order modeling of fluid flows. *Physics of Fluids 31*, 8 (2019), 085101.

[228] Peherstorfer, B., Butnaru, D., Willcox, K., and Bungartz, H.-J. Localized discrete empirical interpolation method. *SIAM Journal on Scientific Computing 36*, 1 (2014), A168–A192.

[229] Peherstorfer, B., Zimmer, S., Zenger, C., and Bungartz, H.-J. A multigrid method for adaptive sparse grids. *SIAM Journal on Scientific Computing 37*, 5 (2015), S51–S70.

[230] Penzl, T. A cyclic low-rank smith method for large sparse lyapunov equations. *SIAM Journal on Scientific Computing 21*, 4 (1999), 1401–1418.

[231] Phillips, J. R. Projection frameworks for model reduction of weakly nonlinear systems. In *Proceedings of the 37th Annual Design Automation Conference* (2000), pp. 184–189.

[232] Phillips, J. R. Projection-based approaches for model reduction of weakly nonlinear, time-varying systems. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 22*, 2 (2003), 171–187.

[233] Pillage, L. T., and Rohrer, R. A. Asymptotic waveform evaluation for timing analysis. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 9*, 4 (1990), 352–366.

[234] Prud'Homme, C., Rovas, D. V., Veroy, K., Machiels, L., Maday, Y., Patera, A. T., and Turinici, G. Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods. *J. Fluids Eng. 124*, 1 (2002), 70–80.

[235] Qi, J., Wang, J., Liu, H., and Dimitrovski, A. D. Nonlinear model reduction in power systems by balancing of empirical controllability and observability covariances. *IEEE Transactions on Power Systems 32*, 1 (2016), 114–126.

[236] Quarteroni, A., Manzoni, A., and Negri, F. *Reduced basis methods for partial differential equations: an introduction*, vol. 92. Springer, 2015.

[237] Rafiq, D., and Bazaz, M. A. A comprehensive scheme for fast simulation of burgers' equation. In *2019 Sixth Indian Control Conference (ICC)* (2019), IEEE, pp. 397–402.

[238] Rafiq, D., and Bazaz, M. A. Model order reduction of non-linear transmission lines using non-linear moment matching. In *2019 International Conference on Computing, Power and Communication Technologies (GUCON)* (2019), IEEE, pp. 394–399.

[239] Rafiq, D., and Bazaz, M. A. A comprehensive scheme for reduction of nonlinear dynamical systems. *International Journal of Dynamics and Control 8*, 2 (2020), 361–369.

[240] Rafiq, D., and Bazaz, M. A. A framework for parametric reduction in large-scale nonlinear dynamical systems. *Nonlinear Dynamics 102*, 3 (2020), 1897–1908.

[241] Rafiq, D., and Bazaz, M. A. Nonlinear model order reduction via nonlinear moment matching with dynamic mode decomposition. *International Journal of Non-Linear Mechanics 128* (2020), 103625.

[242] Rafiq, D., and Bazaz, M. A. Efficient computation of 1d and 2d nonlinear viscous burgers' equation. *International Journal of Dynamics and Control* (2021), 1–13.

[243] Rafiq, D., and Bazaz, M. A. Model order reduction via moment-matching: A state-of-the-art review. *Archives of Computational Methods in Engineering* (2021), 1–21.

[244] Ramaswamy, G., Verghese, G. C., Rouco, L., Vialas, C., and DeMarco, C. Synchrony, aggregation, and multi-area eigenanalysis. *IEEE Transactions on Power Systems 10*, 4 (1995), 1986–1993.

[245] Ramirez, A., Mehrizi-Sani, A., Hussein, D., Matar, M., Abdel-Rahman, M., Chavez, J. J., Davoudi, A., and Kamalasadan, S. Application of balanced realizations for model-order reduction of dynamic power system equivalents. *IEEE Transactions on Power Delivery 31*, 5 (2015), 2304–2312.

[246] Reddy, J. N., and Gartling, D. K. *The finite element method in heat transfer and fluid dynamics*. CRC press, 2010.

[247] Reis, T., and Stykel, T. Balanced truncation model reduction of second-order systems. *Mathematical and Computer Modelling of Dynamical Systems 14*, 5 (2008), 391–406.

[248] Reiss, J., Schulze, P., Sesterhenn, J., and Mehrmann, V. The shifted proper orthogonal decomposition: A mode decomposition for multiple transport phenomena. *SIAM Journal on Scientific Computing 40*, 3 (2018), A1322–A1344.

[249] Rewienski, M., and White, J. A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 22*, 2 (2003), 155–170.

[250] Rewieński, M., and White, J. Model order reduction for nonlinear dynamical systems based on trajectory piecewise-linear approximations. *Linear algebra and its applications 415*, 2-3 (2006), 426–454.

[251] Rozza, G., Huynh, D. B. P., and Patera, A. T. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering 15*, 3 (2008), 229–275.

[252] Rugh, W. J. *Nonlinear system theory*. Johns Hopkins University Press Baltimore, MD, 1981.

[253] Ruhe, A. Rational krylov sequence methods for eigenvalue computation. *Linear Algebra and its Applications 58* (1984), 391–405.

[254] Saad, Y. *Iterative methods for sparse linear systems*. SIAM, 2003.

[255] Sakthivel, R., and Chun, C. New soliton solutions of chaffee-infante equations using the exp-function method. *Zeitschrift für Naturforschung A 65*, 3 (2010), 197–202.

[256] Salimbahrami, B. *Structure Preserving Order Reduction of Large Scale Second Order Models*. PhD thesis, Technical University of Munich, Germany, 2005.

[257] Salimbahrami, B., and Lohmann, B. Order reduction of large scale second-order systems using krylov subspace methods. *Linear Algebra and its Applications 415*, 2-3 (2006), 385–405.

[258] Sandberg, H., and Rantzer, A. Balanced truncation of linear time-varying systems. *IEEE Transactions on Automatic Control 49*, 2 (2004), 217–229.

[259] Sassano, M., and Astolfi, A. Dynamic approximate solutions of the hj inequality and of the hjb equation for input-affine nonlinear systems. *IEEE Transactions on Automatic Control 57*, 10 (2012), 2490–2503.

[260] Sassano, M., and Astolfi, A. Dynamic generalized controllability and observability functions with applications to model reduction and sensor deployment. *Automatica 50*, 5 (2014), 1349–1359.

[261] Sastry, S. *Nonlinear systems: analysis, stability, and control*, vol. 10. Springer Science & Business Media, 2013.

[262] Scarciotti, G. Low computational complexity model reduction of power systems with preservation of physical characteristics. *IEEE Transactions on Power Systems 32*, 1 (2016), 743–752.

[263] Scarciotti, G., and Astolfi, A. Model reduction of neutral linear and nonlinear time-invariant time-delay systems with discrete and distributed delays. *IEEE Transactions on Automatic Control 61*, 6 (2015), 1438–1451.

[264] Scarciotti, G., and Astolfi, A. Data-driven model reduction by moment matching for linear and nonlinear systems. *Automatica 79* (2017), 340–351.

[265] Scarciotti, G., and Astolfi, A. Nonlinear model reduction by moment matching. *Foundations and Trends in Systems and Control* (2017).

[266] Scherpen, J. M., and Gray, W. S. Minimality and local state decompositions of a nonlinear state space realization using energy functions. *IEEE Transactions on automatic control 45*, 11 (2000), 2079–2086.

[267] Scherpen, J. M. A. Balancing for nonlinear systems. *Systems & Control Letters 21*, 2 (1993), 143–153.

[268] Scherpen, J. M. A., and Van der Schaft, A. Normalized coprime factorizations and balancing for unstable nonlinear systems. *International Journal of Control 60*, 6 (1994), 1193–1222.

[269] Schmid, P. J. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics 656* (2010), 5–28.

[270] Schmid, P. J. Application of the dynamic mode decomposition to experimental data. *Experiments in fluids 50*, 4 (2011), 1123–1130.

[271] Shokoohi, S., Silverman, L., and Van Dooren, P. Linear time-variable systems: Balancing and model reduction. *IEEE Transactions on Automatic Control 28*, 8 (1983), 810–822.

[272] Sjöberg, J., Fujimoto, K., and Glad, T. Model reduction of nonlinear differential-algebraic equations. *IFAC Proceedings Volumes 40*, 12 (2007), 176–181.

[273] Son, N. T. A real time procedure for affinely dependent parametric model order reduction using interpolation on grassmann manifolds. *International Journal for Numerical Methods in Engineering 93*, 8 (2013), 818–833.

[274] Son, N. T., and Stykel, T. Solving parameter-dependent lyapunov equations using the reduced basis method with application to parametric model order reduction. *SIAM Journal on Matrix Analysis and Applications 38*, 2 (2017), 478–504.

[275] Sorensen, D. C. Numerical methods for large eigenvalue problems. *Acta Numerica 11* (2002), 519–584.

[276] Sorensen, D. C. Passivity preserving model reduction via interpolation of spectral zeros. *Systems & Control Letters 54*, 4 (2005), 347–360.

[277] Sorensen, D. C., and Antoulas, A. The sylvester equation and approximate balanced reduction. *Linear Algebra and its Applications 351* (2002), 671–700.

[278] Spanos, J. T., Milman, M. H., and Mingori, D. L. A new algorithm for l2 optimal model reduction. *Automatica 28*, 5 (1992), 897–909.

[279] Stewart, G. W. On the early history of the singular value decomposition. *SIAM Review 35*, 4 (1993), 551–566.

[280] Sturk, C., Vanfretti, L., Chompoobutrgool, Y., and Sandberg, H. Coherency-independent structured model reduction of power systems. *IEEE Transactions on Power Systems 29*, 5 (2014), 2418–2426.

[281] Stykel, T. *Analysis and numerical solution of generalized Lyapunov equations*. PhD thesis, Institut für Mathematik, Technische Universität, Berlin, 2002.

[282] Stykel, T. Numerical solution and perturbation theory for generalized lyapunov equations. *Linear Algebra and its Applications 349*, 1-3 (2002), 155–185.

[283] Stykel, T. Gramian-based model reduction for descriptor systems. *Mathematics of Control, Signals and Systems 16*, 4 (2004), 297–319.

[284] Su, T.-J., and Craig Jr, R. R. Model reduction and control of flexible structures using krylov vectors. *Journal of guidance, control, and dynamics 14*, 2 (1991), 260–267.

[285] Swischuk, R., Mainini, L., Peherstorfer, B., and Willcox, K. Projection-based model reduction: Formulations for physics-based machine learning. *Computers & Fluids 179* (2019), 704–717.

[286] Tiso, P., and Rixen, D. J. Discrete empirical interpolation method for finite element structural dynamics. In *Topics in Nonlinear Dynamics, Volume 1*. Springer, 2013, pp. 203–212.

[287] Troullinos, G., Dorsey, J., Wong, H., and Myers, J. Reducing the order of very large power system models. *IEEE Transactions on Power Systems 3*, 1 (1988), 127–133.

[288] Tu, J. H. *Dynamic mode decomposition: Theory and applications*. PhD thesis, Princeton University, 2013.

[289] Tyuryukanov, I., Popov, M., Van Der Meijden, M. A., and Terzija, V. Slow coherency identification and power system dynamic model reduction by using orthogonal structure of electromechanical eigenvectors. *IEEE Transactions on Power Systems* (2020).

[290] Undrill, J., and Turner, A. Construction of power system electromechanical equivalents by modal analysis. *IEEE Transactions on Power Apparatus and Systems*, 5 (1971), 2049–2059.

[291] Uyemura, J. P. *Introduction to VLSI circuits and systems*. Wiley India, 2002.

[292] Van Dooren, P. Numerical linear algebra techniques for large scale matrix problems in systems and control. In *[1992] Proceedings of the 31st IEEE Conference on Decision and Control* (1992), IEEE, pp. 1933–1938.

[293] Van Dooren, P., Gallivan, K. A., and Absil, P.-A. $\mathcal{H}_2$-optimal model reduction of mimo systems. *Applied Mathematics Letters 21*, 12 (2008), 1267–1273.

[294] Varona, M. C., Lohmann, B., et al. Automatic adaptive sampling in parametric model order reduction by matrix interpolation. In *2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)* (2017), IEEE, pp. 472–477.

[295] Varona, M. C., Nico, S., and Lohmann, B. Nonlinear moment matching for the simulation-free reduction of structural systems. In *IFAC Mechatronics and NolCoS, Vienna, Austria* (2019), vol. 52, IFAC, pp. 328 – 333.

[296] Varona, M. C., Raphael, G., Julian, S., and Lohmann, B. Practicable simulation-free model order reduction by nonlinear moment matching, 2019.

[297] Vasilyev, D., Rewienski, M., and White, J. A TBR-based trajectory piecewise-linear algorithm for generating accurate low-order models for nonlinear analog circuits and MEMS. In *Proceedings 2003. Design Automation Conference (IEEE Cat. No. 03CH37451)* (2003), IEEE, pp. 490–495.

[298] Verhulst, F. *Nonlinear differential equations and dynamical systems*. Springer Science & Business Media, 2006.

[299] Veroy, K., and Patera, A. Certified real-time solution of the parametrized steady incompressible navier–stokes equations: rigorous reduced-basis a posteriori error bounds. *International Journal for Numerical Methods in Fluids 47*, 8-9 (2005), 773–788.

[300] Veroy, K., Prud'Homme, C., Rovas, D., and Patera, A. A posteriori error bounds for reduced-basis approximation of parametrized noncoercive and nonlinear elliptic partial differential equations. In *16th AIAA Computational Fluid Dynamics Conference* (2003), p. 3847.

[301] Wang, J. M., Kuh, E. S., and Yu, Q. Passive model order reduction algorithm based on chebyshev expansion of impulse response of interconnect networks. In *Design Automation Conference* (2000), IEEE Computer Society, pp. 520–525.

[302] Wang, S., Lu, S., Zhou, N., Lin, G., Elizondo, M., and Pai, M. Dynamic-feature extraction, attribution, and reconstruction (dear) method for power system model reduction. *IEEE transactions on power systems 29*, 5 (2014), 2049–2059.

[303] Washabaugh, K., Amsallem, D., Zahr, M., and Farhat, C. Nonlinear model reduction for cfd problems using local reduced-order bases. In *42nd AIAA Fluid Dynamics Conference and Exhibit* (2012), p. 2686.

[304] Wedin, P. Å. On angles between subspaces of a finite dimensional inner product space. In *Matrix Pencils*. Springer, 1983, pp. 263–285.

[305] Willcox, K. Unsteady flow sensing and estimation via the gappy proper orthogonal decomposition. *Computers & Fluids 35*, 2 (2006), 208–226.

[306] Willcox, K., and Peraire, J. Balanced model reduction via the proper orthogonal decomposition. *AIAA journal 40*, 11 (2002), 2323–2330.

[307] Winkelman, J., Chow, J., Bowler, B., Avramovic, B., and Kokotovic, P. An analysis of interarea dynamics of multi-machine systems. *IEEE Transactions on Power Apparatus and Systems*, 2 (1981), 754–763.

[308] Wirtz, D., Sorensen, D. C., and Haasdonk, B. A posteriori error estimation for deim reduced nonlinear dynamical systems. *SIAM Journal on Scientific Computing 36*, 2 (2014), A311–A338.

[309] Wolf, T., Panzer, H., and Lohmann, B. Gramian-based error bound in model reduction by krylov subspace methods. *IFAC Proceedings Volumes 44*, 1 (2011), 3587–3592.

[310] Yan, W.-Y., and Lam, J. An approximate approach to $\mathcal{H}_2$ optimal model reduction. *IEEE Transactions on Automatic Control 44*, 7 (1999), 1341–1358.

[311] Younis, M. I., Abdel-Rahman, E. M., and Nayfeh, A. A reduced-order model for electrically actuated microbeam-based mems. *Journal of Microelectromechanical Systems 12*, 5 (2003), 672–680.

[312] Yue, Y., and Meerbergen, K. Accelerating optimization of parametric linear systems by model order reduction. *SIAM Journal on Optimization 23*, 2 (2013), 1344–1370.

[313] Zhao, H. S., Xue, N., and Shi, N. Nonlinear dynamic power system model reduction analysis using balanced empirical gramian. In *Applied Mechanics and Materials* (2014), vol. 448, Trans Tech Publ, pp. 2368–2374.

[314] Žigić, D., Watson, L. T., and Beattie, C. Contragredient transformations applied to the optimal projection equations. *Linear Algebra and its Applications 188* (1993), 665–676.

[315] Zimmerman, R. D., Murillo-Sánchez, C. E., and Thomas, R. J. Matpower: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on power systems 26*, 1 (2010), 12–19.