Local stability guarantees for data-driven quadratically nonlinear models

Mai Peng

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science

University of Washington

2023

Committee:

Steven L. Brunton

Alan A. Kaptanoglu

Krithika Manohar

Program Authorized to Offer Degree:
Mechanical Engineering

University of Washington

Abstract

Local stability guarantees for data-driven
quadratically nonlinear models

Mai Peng

Chair of the Supervisory Committee:
Steven L. Brunton
Department of Mechanical Engineering

Navier Stokes equations (NSEs) are complicated partial differential equations (PDEs) to describe the motion of fluids which are computationally expensive to simulate because of the high dimensionality. Reduced-order models (ROMs) are simpler models for evolving the flows by capturing only the dominant behaviors of a system. However it is challenging to guarantee the stability of these models either globally or locally. For quadratically nonlinear systems that represent many fluid flows, there is a theorem about global stability, which can be used to check if such ROMs are globally stable. Next, it was incorporated into a method that determine models directly from data, the sparse identification of nonlinear dynamics (SINDy) method in a modified technique called "trapping SINDy". In this work, we relax the quadratically energy-preserving constraints and promote local stability in data-driven models of quadratically nonlinear dynamics. This is important because this weakly quadratically energy-preserving structure exists in a large number of boundary conditions of fluids. First we raise a theorem outlining the sufficient condition to ensure local stability in linear-quadratic systems and provide an estimate stability radius based on the theorem. Second, we incorporate this theorem into data-driven models and present how we form the optimization problem based on the theorem. Then a modified "extended trapping SINDy" algorithm is introduced based on "trapping SINDy", enabling the trajectories of the data-driven models obtained via this method to be locally bounded inside a given ball-

shape trapping region. Several examples are presented to demonstrate the effectiveness and accuracy of the proposed algorithm.

TABLE OF CONTENTS

Page

LIST OF FIGURES

ACKNOWLEDGMENTS

First I would like to thank the committee chair, Prof. Steve Brunton, who provided me the opportunity to be part of this wonderful research group, dynamics lab at University of Washington. I have enjoyed everyday working with a group of talents who are warm-hearted and always willing to help. Without Prof. Brunton's support, I would not have the chance to utilize what I have learned throughout the program to conduct such an interesting research work.

I must thank Prof. Alan Kaptanoglu for his invaluable advice, guidance and encouragement which gives me the confidence to achieve something from this study. I have benefited a lot from his patience and erudition in this topic, which always guides me in the right direction. I here want to also congratulate his new professorship job in New York. He definitely deserves it and I extend my best wishes for the next chapter of his academic career.

I would also like to thank Dr. Chris Hansen for his initial support of this research and advice from start to end, which turns me from a graduate student to a graduate researcher.

I am thankful to Prof. Krithika Manohar who has served as one of the Master committee members, readers and examiners, who expands the scope of my study at University of Washington.

And finally I want to thank all my groupmates in dynamics lab, my friends and my family.

DEDICATION

In memory of my grandmother,

Zou Xiangzhen

Chapter 1

INTRODUCTION

1.1 Overview and motivation

A fluid is the material that continuously deforms under an applied shear stress or external force such as liquid and gas. From the sketches of Leonardo Da Vinci to George Caley's conception of a modern airplane configuration, and from Archimedes' *On Floating Bodies* to the works on viscous fluid substances by Claude-Louis Navier and George Gabriel Stokes, there is a long history of human trying to understand and make use of the fluids. In spite of the fact that the underlying governing equations of viscous fluid flow being known for hundred years, there is much yet to learn and discover. One classical example is Navier-Stokes equations, which are complicated partial differential equations (PDEs) to describe the motion of fluids which are computationally expensive to simulate because of the high dimensionality. In fact, modeling the full multi-scale, spatio-temporal evolution of most dynamical systems is generally computationally expensive, which motivates the use of reduced-order models (ROMs). ROMs are simpler models for evolving the flows by capturing only the dominant behaviors of a system[37, 38, 9, 10, 3, 44]. Projection-based model reduction is a common approach for generating ROMs. With this technique, a high-dimensional system, such as a spatially discretized set of partial differential equations (PDEs), is projected onto a lower-dimensional basis of modes [55, 56]. The projection leads to a computationally efficient system of ordinary differential equations (ODEs) that describes how the mode amplitudes evolve in time [20]. One example of the ROMs is orthogonal projection onto proper orthogonal decomposition (POD) modes, which results in POD-Galerkin models. However, there is no guarantee that the method is globally or locally stable, and the stability issues often cause solutions to diverge in finite-time. The traditional explanation is that the stability issues in Galerkin models derive from truncated dissipative scales, but increasingly there are alternate explanations including fundamental numerical issues with the Galerkin framework,

potentially resolved in a Petrov-Galerkin framework, for convection-dominated flows [17]. To obtain reliable models motivates methods to build computationally inexpensive ROMs whose stability can be trusted.

To determine if a Galerkin model is long-term bounded, Schlegel and Noack [48] developed a "trapping theorem" with necessary and sufficient conditions for long-term model stability for systems that exhibit quadratic, energy-preserving nonlinearities. This theorem can be used as an effective diagnostic for evaluating the stability of reduced-order models. Moreover, the theorem was subsequently incorporated into system identification methods, for producing data-driven models that are provably long-term bounded [22]. The ability to promote global stability guarantees has broad potential for data-driven models, and has already been extended to neural-network-based system identification methods [39].

The assumptions of the trapping theorem may seem somewhat limiting, but actually there are many scenarios under which energy-preserving quadratic nonlinearities can arise. In fluid dynamics, the quadratic nonlinearity often represents the convective derivative $(\boldsymbol{u} \cdot \nabla)\boldsymbol{u}$ in the Navier-Stokes equations (NSEs). This quadratic nonlinearity is energy-preserving for a large number of boundary conditions. Examples include no-slip conditions, periodic boundary conditions [34, 20], mixed no-slip and periodic boundary conditions [45], and open flows in which the velocity magnitude decreases faster than the relevant surface integrals expand (e.g., two-dimensional rigid body wake flows and three-dimensional round jets) [49]. In magnetohydrodynamics (MHD), and MHD extensions such as Hall-MHD, there are additional quadratic nonlinearities that are also energy-preserving with common experimental boundary conditions such as a conducting wall [15], or a balance between dissipation and actuation in a steady-state plasma device [21, 24].

The following question may arise for dynamical systems that do not hold the energy-preserving structure: how can we find some basic local stability guarantees for such systems? For general dynamical systems especially for those with large degrees of freedom, it seems not possible to find the domain where trajectories that start at large distance of the origin are bounded. However, we have noticed that many fluid flows, i.e. those with open channels on part of the boundary, have this weak breaking of the energy-preserving symmetry. Hence, in this thesis, we focus on the quadratically nonlinear systems and seek local stability

guarantees for such systems by relaxing the constraints of energy-preserving nonlinearities.

1.2 Data-driven modeling

Data has been an essential part of scientific research for a long time. The most common usage of the data might be to verify the hypothesis or theory. However, the presence of increasingly large data sets motivates the use of data for modeling. Although natural philosophers, mathematicians and physicists in history, started to develop data-driven models centuries ago, they often did not capture the dynamic relationships behind the mathematical discoveries, for example, the famous elliptic orbits of planetary motion by Kepler. Therefore, appropriate data analysis techniques are needed to help researchers extract physically relevant information from the data. A breakthrough of the new approach to find nonlinear differential equations using symbolic regression, has been made by Bongard and Lipson [4] and Schmidt and Lipson [50]. With the rapid development of the computational power in the past few decades, there has been a tremendous leap forward in statistical and machine learning techniques which now become powerful approaches for data-driven modeling as well.

For complex systems that are generally nonlinear, such as fluids, it is favorable to build simple models from the abundant data through tractable data analysis techniques, which can hopefully further predict the long-time behavior of the dynamics. A much-desired goal is to obtain lower-dimensional models from the high-dimensional data sets, which can benefit the fundamental understanding of such systems and significantly speed up numerical predictions. It is also expected that we may able to predict and control the systems by the use of lower-dimensional data-driven models as well. Even without being able to accurately identify, predict or control the dynamics with the lower-dimensional model, the lower-dimensional information about the system such as spatial modes corresponding to certain energetic or dynamic characteristics, can still be important for improving the fundamental understanding of the system. Combining the data-driven modal decomposition or other data-driven methods, and theoretical foundation of the system could potentially produce the desired lower-dimensional models.

An important data-driven method, utilized later in this work to build data-driven ROMs

of fluid flow, is the sparse identification of nonlinear dynamics (SINDy) by Brunton *et al.* [7]. The SINDy is an algorithm that can uncover the governing equations of a system from data. This low dimensional representation is an input to the SINDy method which utilizes a predefined function library to uncover the sparse representation of the dynamics using a regularizer such as LASSO [57] or least squares thresholding.

Benefiting from the ever-increasing rise of the computational power, machine learning approaches for data-driven techniques such as deep learning are now widely used in science and engineering. Deep learning has been applied to dynamical systems in many ways for purposes of modeling and control [59]. It encompasses the potential to learn a model for complex dynamics that may not have analytical forms or to uncover new relationships that were previously undiscovered. Deep learning comes with many network hyperparameters, such as network structure and network loss function that may drastically impact the learned function. An example of the successful use of deep learning based on SINDy to learn dynamical systems is the SINDy autoencoders by Champion *et al.* [11].

1.3 Organization and contributions

In this thesis, we develop a local stability theorem in the context of projection-based ROMs. Additionally, it enables us to promote local stability of data-driven models with quadratic nonlinearities. The theorem provides sufficient conditions under which quadratically nonlinear systems with a relaxed quadratically energy-preserving structure are locally stable. By tightening the relaxation, the theorem reproduces the seminal Schlegel and Noack theorom [48, 22] related to global stability. We next present how to promote the local stability in data-driven models with a modified optimization loss function that is designed under the guidance of this theorem. The PySINDy code [23] provides a number of useful readily available SINDy-based tools and is an ideal platform for illustrating this approach and implementing our custom optimization loss terms.

Following this introductory chapter, Chap. 2 presents a summary of a few key concepts related to the main structure of this thesis. In the same chapter, we review necessary background knowledge to help readers quickly catch the core essence of this work. The problem under consideration in this thesis is defined in Sec. 3.1. Next, we introduce the Schlegel

and Noack trapping theorem, which connects the stability of the system and the energy-preserving structure introduced in Sec. 3.2. We then present a new theorem which derives an analytical estimates for the stability domain under the situation where the energy-preserving constraints are relaxed. In Chap. 4, we define our "extended trapping SINDy" algorithm to promote local stability of data-driven models. The implementation is available through the PySINDy software package which is open-source [13]. In Sec. 4.2, we demonstrate the efficacy and utility of this new technique for system identification and modeling on a number of examples including data obtained from a noisy Lorenz attractor, the canonical lid-driven cavity flow, and assorted locally stable chaos systems from the open-source dysts database [16]. In Chap. 5, we conclude our contribution and main achievement of this work, which is that we give practitioners guarantees about stability of data-driven models even when the energy-preserving constraints are broken. We then suggests some of the future work at the end.

Chapter 2
BACKGROUND

This chapter will present several key concepts that will be utilized and referenced repeatedly throughout this thesis. Sec. 2.1 introduces widely employed algorithms for model identification and simulation of complex systems. Comprehensive analysis of relevant applications of this method was previously presented in detail in Sec. 1.1. The method predominantly utilized to construct the problem discussed in Chap. 3 in this thesis is the POD-Galerkin method, which will be divided into POD and Galerkin expansion to be seperately introduced in Sec. 2.1.1 and Sec. 2.1.2. We perform stability analysis on our data-driven model using the Lyapunov's direct method in Chap. 3. A brief introduction of Lyapunov stability will be presented in Sec. 2.3. And a main data-driven algorithm used by our work will be introduced in Sec. 2.2.1. Finally, underscoring the principal physical feature of the systems we study, Sec. 3.2 will introduce flows exhibiting energy-preserving nonlinearities.

2.1 Reduced-order modeling

The classical methods of constructing models of physical behaviors are to take the advantage of analytical techniques that has been continuously developed over past centuries. Compared to the analytical methods, numerical methods have relatively short history and was only practical for simulations with small numbers of degrees of freedom when it was introduced at the very beginning. Thanks to the recent appearance of powerful computational platforms, it is possible now to achieve high-fidelity models by adopting computational-based methods, even for problems with millions of degrees of freedom. However, without the help of other analysis or tools, addressing complex systems by the use of numerical simulation is still commonly challenging in that it often does not improve the fundamental understanding of the physics [33]. For example, partial differential equations (PDEs) can be discretized using finite-difference differentiation schemes which transform the governing PDE into a

high-dimensional set of ordinary differential equations (ODEs). Although detailed time histories of spatial points from discretization can be collected using such method, the lack of low-order description of the physical behaviors causes difficulties in desired tasks such as system prediction and control.

To provide a more accurate depiction of the challenges involved in addressing complex systems through numerical methods, consider a system of generic nonlinear PDEs in a one-dimensional space,

$$\boldsymbol{u}_t = \mathbf{N}(\boldsymbol{u}_x, \boldsymbol{u}_{xx}, \cdots, x, t) \tag{2.1}$$

where the operator $\mathbf{N}$ prescribes the nonlinear evolution and the number of subscripts indicate the partial derivatives are of the corresponding order. Although there are a few analytical methods such as the separation of variables, to solve PDEs, nonlinear PDEs do not have closed form solutions in general. However, the philosophy of separation of variables provides a feasible direction to address the challenges posed by PDEs, which is to reduce the PDE to a set of ODEs. The underlying cause for this transformation is that there is a more extensive repertoire of analytical methodologies for studying the nonlinear behavior of ODEs [53]. With the help of computational techniques, high-dimensional numerical solutions can be achieved approximately using various newly developed numerical tools. To be more specific, consider the spatial descretization of the spatial variable x evaluated at $n \gg 1$ points where $x_k \in [-L, L]$ for $k = 1, 2, \cdots, n$. At each point we have $\boldsymbol{u}(x_k, t)$ with spacing $\Delta x = x_{k+1} - x_k = 2L/n$. And the spatial derivatives can be given by the second-order accurate center-difference schemes,

$$\boldsymbol{u}_x = \frac{\boldsymbol{u}(x_{k+1}, t) - \boldsymbol{u}(x_{k-1}, t)}{2\Delta x} \tag{2.2a}$$

$$\boldsymbol{u}_{xx} = \frac{\boldsymbol{u}(x_{k+1}, t) - 2\boldsymbol{u}(x_k, t) + \boldsymbol{u}(x_{k-1}, t)}{\Delta x^2} \tag{2.2b}$$

The discretization transforms the PDE 2.1 into a set of ODEs below,

$$\dot{\boldsymbol{u}}_k = \mathbf{N}\left(\boldsymbol{u}(x_{k+1}, t), \boldsymbol{u}(x_k, t), \boldsymbol{u}(x_{k-1}, t), \cdots, x_k, t\right), \quad k = 1, 2, \cdots, n. \tag{2.3}$$

It should be noted that the n here is often very large. Hence although the system of ODEs seems to be more manageable than the original PDEs, directly solving it with numerical

methods may require considerable computing capability. Furthermore, to increase the accuracy of the solution with the same differential schemes, typically the spacing Δx should be decreased, which, on the contrary, increase the dimension n of the system. This issue with computational speed can be ameliorated if the physical system in question exhibits some underlying dynamical structures that dominate the evolution of the flow. And potentially we can utilize these structures to build low-rank models that can well simulate the original governing PDEs are desirable for improvements in computational speed. A widely used method in fluid dynamics to produce the low-rank models is the proper orthogonal decomposition (POD).

2.1.1 Proper orthogonal decomposition

The proper orthogonal decomposition (POD) [20, 6] is an algorithm based on singular value decomposition (SVD) applied to PDEs, whose goal is to obtain a set of empirical spatial modes that effectively capture the energy distribution within a given dataset. Although as previous mentioned, separation of variables cannot be used to solve nonlinear PDEs analytically, it can still exhibit strong performance in computational methods, yielding accurate solutions through appropriate algorithms. Assume that the dynamics of Eq. (2.1) can be decomposed as,

$$\boldsymbol{u}(x,t) = \sum_{i=1}^{n} \boldsymbol{\chi}_i(\boldsymbol{x}) a_i(t), \tag{2.4}$$

where $\boldsymbol{\chi}_i$ are a set of $n \gg 1$ basis modes. Common choices of basis for such expansion include Fourier mode expansion and POD. For POD basis, consider a collection of snapshots consisting of samples of a complex system,

$$\boldsymbol{Y} = \begin{bmatrix} | & | & & | \\ \boldsymbol{u}_1 & \boldsymbol{u}_2 & \cdots & \boldsymbol{u}_m \\ | & | & & | \end{bmatrix} \tag{2.5}$$

where each snapshot $\boldsymbol{u}_k$ are sampled at time t_k so that $\boldsymbol{u}_k = [u(x_1,t_k)\ \ u(x_2,t_k)\ \cdots\ u(x_n,t_k)]^T$ and $\boldsymbol{u}_k \in \mathbb{R}^n$. Typically we have $n \gg m \gg 1$, which makes $\boldsymbol{Y}$ a tall-skinny high-dimensional matrix. And SVD can provide a unique decomposition for $\boldsymbol{Y}$ as,

$$\boldsymbol{Y} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* \tag{2.6}$$

where $\boldsymbol{U} \in \mathbb{C}^{n\times n}$ is a unitary matrix and the nonnegative i^{th} entry on the diagonal of the matrix $\boldsymbol{\Sigma} \in \mathbb{C}^{n\times m}$ called singular values which correspond to the energy contained in the i^{th} mode and are ordered from largest to smallest. Then the POD modes are the columns of $\boldsymbol{U}$, which are called left singular vectors of $\boldsymbol{Y}$ as well. One reason that POD modes can form the ideal basis for Eq. (2.4) is that the columns of matrix $\boldsymbol{U}$ are orthonormal. This orthogonality properties allow us to take the advantage of the method of eigenfunction expansions to produce reliable solutions. That is, for $\boldsymbol{u}_i$ and $\boldsymbol{u}_j$ being columns of $\boldsymbol{U}$, we have

$$\langle \boldsymbol{\chi}_i, \boldsymbol{\chi}_j \rangle_{\boldsymbol{W}} = \boldsymbol{\chi}_j^* \boldsymbol{W} \boldsymbol{\chi}_i = \delta_{ij} \tag{2.7}$$

where $\langle \cdot, \cdot \rangle$ denotes the usual inner product, $\boldsymbol{W}$ is an appropriate weight matrix, δ_{ij} is the Kronecker delta function and * denotes the conjugate transpose.

If matrix $\boldsymbol{Y}$ does not have full rank, the SVD of $\boldsymbol{Y}$ will yield the $\boldsymbol{\Sigma}$ with multiple singular values that are zero to machine precision. This implies that it is feasible to truncate the $\boldsymbol{\Sigma}$, resulting in a truncated SVD that can still be exact. Similarly, given a certain positive integer $r \ll n$, a reduced-order approximation of the original data $\boldsymbol{Y}$ can be achieved by simply truncating the columns of $\boldsymbol{U}$ beyond r. Then we have the rank-r approximation of $\boldsymbol{Y}$,

$$\tilde{\boldsymbol{Y}} = \boldsymbol{U}_r \boldsymbol{\Sigma}_r \boldsymbol{V}_r^* \tag{2.8}$$

where $\boldsymbol{U}_r$ and $\boldsymbol{V}_r$ contains first r columns of $\boldsymbol{U}$ and $\boldsymbol{V}$, and $\Sigma \in \mathbb{R}^{r\times r}$ contains the largest r singular values. In other words, some modes that are of sufficiently low energy (that is, small singular values) are removed. It is important to keep in mind that in order to preserve the r most dominant modes that capture the major dynamics, it is typically necessary to ensure that $\|\boldsymbol{Y} - \tilde{\boldsymbol{Y}}\| < \epsilon$, for a specified small value ϵ.

2.1.2 Galerkin projection

The concept of galerkin projection is to take the advantage of the orthogonality properties of the basis functions to make use of Eq. (2.4) so that the governing PDEs can be approximated by a low-dimensional set of ODEs. To be more specific, the equations (2.1) are projected

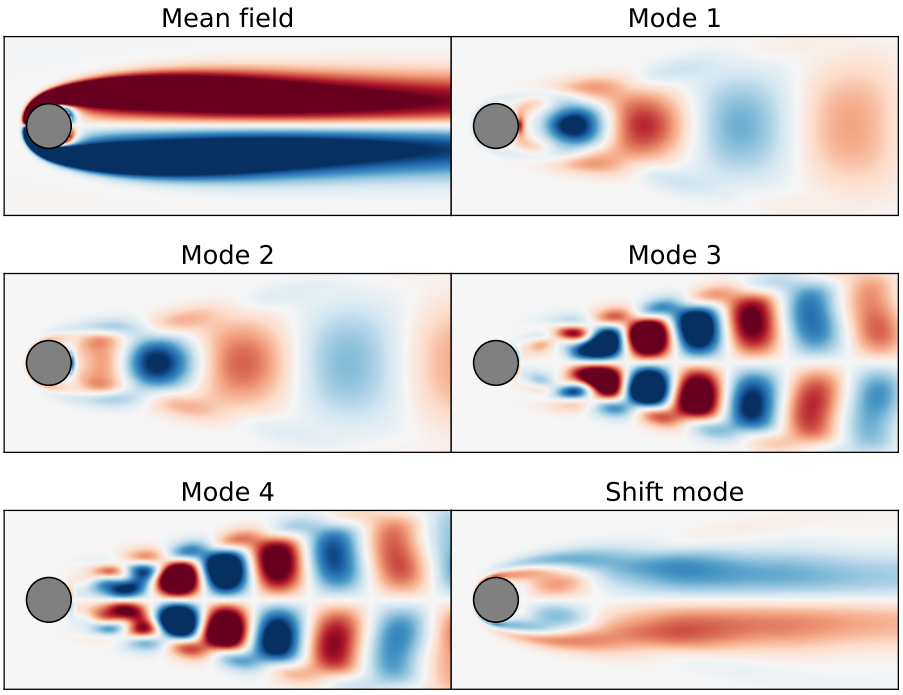

Figure 2.1: Snapshots of von Karman vortex shedding on first 4 POD modes + shift modes.

onto expansions of the form (2.4) to get,

$$\sum \boldsymbol{\chi}_i \frac{da_i}{dt} = \mathbf{N}\left(\sum a_i \boldsymbol{\chi}_i, \sum a_i (\boldsymbol{\chi}_i)_x, \sum a_i (\boldsymbol{\chi}_i)_{xx}, \cdots, x, t\right). \tag{2.9}$$

Then we take the inner product of both sides of the equation and any mode $\boldsymbol{\chi}_j$, and then use the orthogonality properties to get,

$$\frac{da_j}{dt} = \left\langle \mathbf{N}\left(\sum a_i \boldsymbol{\chi}_i, \sum a_i (\boldsymbol{\chi}_i)_x, \sum a_i (\boldsymbol{\chi}_i)_{xx}, \cdots, x, t\right), \boldsymbol{\chi}_j \right\rangle \quad k = 1, 2, \cdots, n. \tag{2.10}$$

We might not see the advantage of Galerkin projection here now, however we successfully achieve a low-dimensional set of ODEs from the original PDEs in this way. Indeed, there are many commonly used numerical schemes are based on Galerkin projection to simulate

the full governing PDEs like (2.1). Although we previously mentioned that the strategy of separation of variables for analytically solving PDEs does not work for nonlinear systems, it may work well with numerical methods for nonlinear and nonconstant coefficient PDEs, if there are enough modal basis functions to capture all the nonlinear mode mixing in (2.10), which comes from the mode-coupling.

2.1.3 Galerkin models of fluid flows

In this subsection, we perform the POD-Galerkin projection onto the Navier-Stokes equation to illustrate the effectiveness of this method. Agian, the state of the system $\boldsymbol{u}(\boldsymbol{x},t) \in \mathbb{R}^n$ is defined as a high-dimensional vector that represents the fluid velocity or other set of spatio-temporal fields, for example sampled on a high-resolution spatial grid. For concreteness in the fluid examples in this work, we will assume that the dynamical flow is evolving according to the incompressible Navier-Stokes equations,

$$\frac{\partial \boldsymbol{u}}{\partial t} = -\boldsymbol{u} \cdot \nabla \boldsymbol{u} + \nu \nabla^2 \boldsymbol{u} - \frac{1}{\rho} \nabla p, \tag{2.11}$$

The density ρ and viscosity ν are assumed constant, and the pressure is denoted p. These equations are expected to well approximate incompressible flow of Newtonian fluids with constant density and kinematic viscosity such as lid-driven cavity flow, fluid flow in channel, steady flow in tubes and so on. Other complex fluid flows such as ocean current, viscoelastic fluids or ferrofluids are described by models that produces higher order polynomial nonlinearities or trigonometric nonlinearities, so they cannot be approximated by NSEs.

The goal of a projection-based ROM is to transform this high-dimensional system into a lower-dimensional system of size $r \ll n$ that captures the essential dynamics. The way to reduce the set of governing equations to a set of ordinary differential equations is by decomposition into a desired low-dimensional basis $\{\boldsymbol{\chi}_i(\boldsymbol{x})\}$ in a process referred to as Galerkin expansion:

$$\boldsymbol{u}(\boldsymbol{x},t) = \overline{\boldsymbol{u}}(\boldsymbol{x}) + a_i(t)\boldsymbol{\chi}_i(\boldsymbol{x}). \tag{2.12}$$

The sum over i is implied and $i \in \{1, ..., r\}$. Here, $\boldsymbol{\chi}_i(\boldsymbol{x})$ are spatial modes of the expansion for the fluctuation $\boldsymbol{u}'(\boldsymbol{x},t) = \boldsymbol{u}(\boldsymbol{x},t) - \overline{\boldsymbol{u}}(\boldsymbol{x})$, $a_i(t)$ describe how the amplitude of these modes

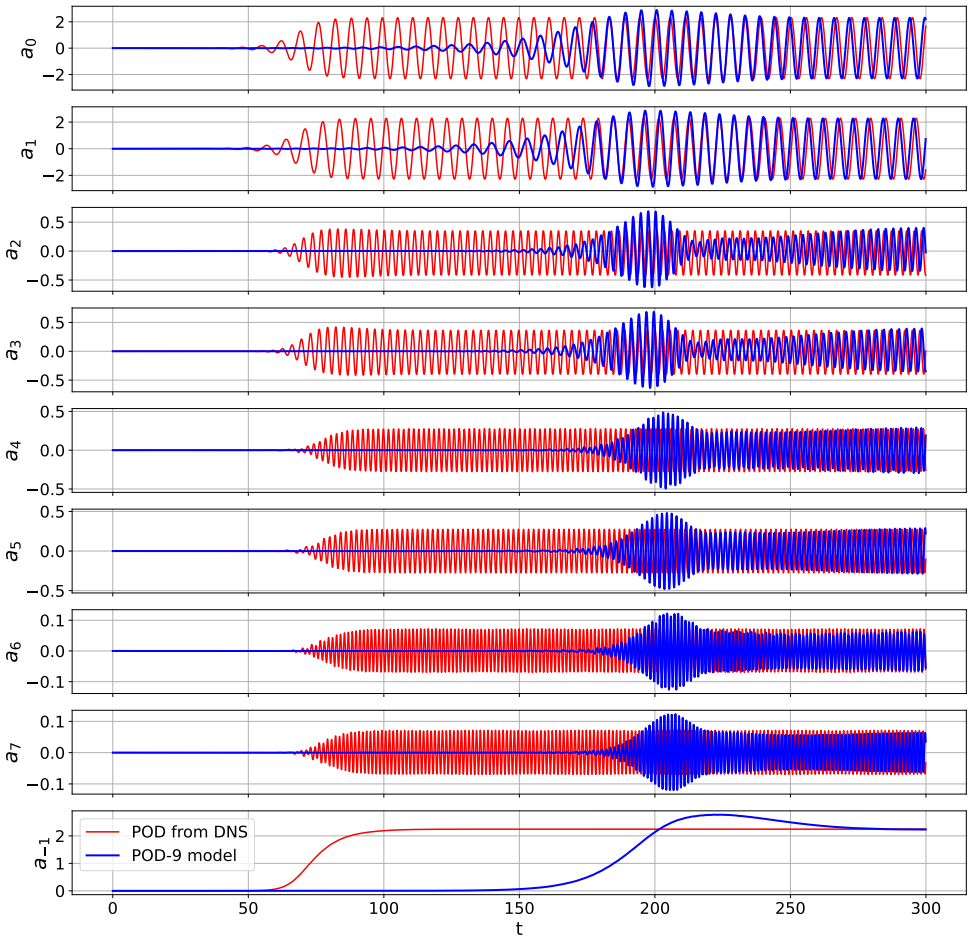

Figure 2.2: DNS and POD-Galerkin models on first 8 POD modes + shift modes.

vary in time, and $\overline{\boldsymbol{u}}(\boldsymbol{x})$ is a base flow which fulfills stationary boundary conditions. We then use proper orthogonal decomposition (POD) to obtain the basis, as shown in Fig. 2.2, since the modes $\boldsymbol{\chi}_i(\boldsymbol{x})$ are orthogonal and ordered by maximal energy content. The governing equations are then Galerkin projected onto the basis $\{\boldsymbol{\chi}_i(\boldsymbol{x})\}$ by substituting Eq. (2.12)

into the PDE and using inner products to remove the spatial dependence, which gives

$$\left\langle \frac{\partial \boldsymbol{u}}{\partial t}, \boldsymbol{\chi}_j \right\rangle = -\left\langle \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{u}, \boldsymbol{\chi}_j \right\rangle + \nu \left\langle \boldsymbol{\Delta} \boldsymbol{u}, \boldsymbol{\chi}_j \right\rangle - \frac{1}{\rho} \left\langle \boldsymbol{\nabla} p, \boldsymbol{\chi}_j \right\rangle, \tag{2.13}$$

which is,

$$\dot{\boldsymbol{a}} = \boldsymbol{E} + \boldsymbol{L}\boldsymbol{a} + \boldsymbol{Q}(\boldsymbol{a}, \boldsymbol{a}), \tag{2.14}$$

where $\boldsymbol{E}$ is a vector representing the bias, $\boldsymbol{L}$ is a linear operator (i.e., a matrix), and $\boldsymbol{Q}$ is a bilinear operator (i.e., a third-order tensor). They are each defined as follows,

$$E_i = -\frac{1}{\rho} \left\langle \boldsymbol{\nabla} p, \boldsymbol{\chi}_i \right\rangle, \tag{2.15a}$$

$$L_{ij} = \nu \left\langle \boldsymbol{\Delta} \boldsymbol{\chi}_j, \boldsymbol{\chi}_i \right\rangle, \tag{2.15b}$$

$$Q_{ijk} = -\left\langle \boldsymbol{\chi}_j \cdot \boldsymbol{\nabla} \boldsymbol{\chi}_k, \boldsymbol{\chi}_j \right\rangle. \tag{2.15c}$$

Orthogonal projection onto POD modes is the simplest and most common procedure, resulting in POD-Galerkin models, although Petrov-Galerkin projection [9, 10] improves model performance in some cases.

2.2 Data-driven modeling of fluid dynamics

In the last section, we introduced several widely used data-driven techniques and algorithms for the study of dynamical systems. In this section, we present another popular methodology which is machine learning for constructing data-driven models in the field of fluid dynamics. The field of fluid mechanics offers a wealth of data and poses numerous challenging problems, making it an ideal test field for the application of machine learning techniques. Machine learning's unprecedented achievements in tasks like computer vision, data mining, and natural language processing, which were traditionally challenging, have captured the attention of scientists and engineers who are now turning to this emerging tool to address complex problems in fluid dynamics. It has shown great promise in enhancing our understanding of fluid dynamics, optimizing designs, and solving complex fluid flow problems [43, 62]. The continuously increasing computational power of modern computers has made machine learning an even more potent tool for optimization tasks in high-dimensional systems such as fluid dynamics than ever before. A fascinating aspect to consider is that

both machine learning and fluid mechanics share a common premise to lift the performance on optimization tasks, which is that they tend to operate under the assumption that exploitable patterns exist [55].

Machine learning is considered as an automatic black box modeling strategy extracting models directly from data, which is not entirely accurate. Instead, applying machine learning to study physical sciences and engineering requires some physics or fundamental principles with data-driven components, which makes it a "gray" box [36]. Moreover, to obtain a reliable machine learning model for fluid dynamics, it requires human intelligence to not only decide on the problem but also select the most appropriate architecture to model the data, design custom loss functions to assess performance quantitatively, lift the performance of the process by implementing specific optimization algorithms and so on [5]. Unlike many applications of machine learning in computer science and engineering, as previously noted, dealing with fluid flows demands the quantification of the underlying physical mechanisms, in most cases, to enable comprehensive analysis, prediction and control. For instance, the identification or simulation of unsteady flows such as the famous von Karman vortex street, relies on customized algorithms to address the nonlinearities and stability issues [60, 41], which may not be part of widely-adopted machine learning algorithms.

In this thesis, we employ supervised learning to identify quadratically nonlinear models from data and promote its local stability using customized loss function based on Thm. 3.2 in Sec. 3.4.1. In supervised learning, the algorithm is trained to make predictions or decisions by establishing a relationship between input data and their corresponding output labels. A commonly used form of loss functions in supervised learning is,

$$L\left(\mathbf{y}, \mathscr{F}\left(\mathbf{x}, \mathbf{y}, \mathbf{w}\right)\right) = \|\mathbf{y} - \mathscr{F}\left(\mathbf{x}, \mathbf{y}, \mathbf{w}\right)\|,$$

where $\mathbf{x}$ and $\mathbf{y}$ represent the input and output (labels) data, $\mathscr{F}$ defines the structure which is controlled by parameters $\mathbf{w}$.

A powerful machine learning tool for dynamics called PySINDy [23] based on the algorithm of sparse identification of nonlinear dynamics (SINDy) [7] is employed to practice our customized loss functions producing results of a few different tasks. Next we will introduce the basic SINDy algorithm and its constrained form.

2.2.1 Sparse identification of nonlinear dynamics

Identifying dynamical systems from data has been a formidable challenge in mathematical physics for a long time, especially in the study of fluid dynamics. The SINDy algorithm leverages the fact that for most physical systems, only a few relevant terms determine the dynamics, to identify parsimonious models from data while avoiding over-fitting.

Calculating the POD-Galerkin model coefficients in Equations (2.15) can be computationally intensive, and the deterministic process provides no way to enforce a notion of stability in this low-dimensional model. Subsequently, we are interested in using the SINDy technique to directly determine the coefficients of POD-Galerkin-like models from data alone; later, we will additionally enforce stability properties in the identified model. To determine the system of Eq. (2.14) from data, we build SINDy models for the dynamics of the temporal POD modes $\boldsymbol{a}(t)$. We assume for parsimony's sake that the governing dynamical equations of $\boldsymbol{a}(t)$ involve only a few terms from the library $\boldsymbol{\Theta}$, consisting of a few candidate functions of the state variable, which makes it sparse in the nonlinear function space. The library $\boldsymbol{\Theta}$ may have nonlinear terms such as polynomial and trigonometric terms. If we consider only polynomial terms to construct the library, then the dynamics will be described as,

$$\frac{d}{dt}\boldsymbol{x} = \boldsymbol{\Theta}(\boldsymbol{x})\boldsymbol{\Xi}, \qquad \boldsymbol{\Theta}(\boldsymbol{x}) = \begin{bmatrix} | & | & | & | & \\ 1 & \boldsymbol{x} & \boldsymbol{x}^{\boldsymbol{P_2}} & \boldsymbol{x}^{\boldsymbol{P_3}} & \cdots \\ | & | & | & | & \end{bmatrix}. \tag{2.16}$$

Here, $\boldsymbol{x}^{\boldsymbol{P_i}}$ denotes polynomials that are formed by the state $\boldsymbol{x}$, where $\boldsymbol{P}_i$ indicate the orders of the polynomials. Consider the collections of snapshots of state $\boldsymbol{x}$ and $\dot{\boldsymbol{x}}$ which come from either measurement or numerical approximation from $\boldsymbol{a}$,

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}(t_1) & \boldsymbol{x}(t_2) & \cdots & \boldsymbol{x}(t_M) \end{bmatrix}^T,$$
$$\dot{\boldsymbol{X}} = \begin{bmatrix} \dot{\boldsymbol{x}}(t_1) & \dot{\boldsymbol{x}}(t_2) & \cdots & \dot{\boldsymbol{x}}(t_M) \end{bmatrix}^T.$$

The dynamics of Eq. (2.2.1) becomes $\dot{\boldsymbol{X}} = \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi}$, where $\boldsymbol{\Xi} = [\boldsymbol{\xi}_1\ \boldsymbol{\xi}_2\ \cdots\ \boldsymbol{\xi}_n]$ is a sparse matrix determined by SINDy whose nonzero elements in each $\boldsymbol{\xi}_j$ indicates that corresponding candidate functions in the library $\boldsymbol{\Theta}$ are active. For instance, the $\boldsymbol{X}^{\boldsymbol{P}_2}$ should be as

follows,

$$\boldsymbol{X}^{\boldsymbol{P}_2} = \begin{bmatrix} \boldsymbol{x}_1^2(t_1) & \boldsymbol{x}_1(t_1)\boldsymbol{x}_2(t_1) & \cdots & \boldsymbol{x}_2^2(t_1) & \cdots & \boldsymbol{x}_n^2(t_1) \\ \boldsymbol{x}_1^2(t_2) & \boldsymbol{x}_1(t_2)\boldsymbol{x}_2(t_2) & \cdots & \boldsymbol{x}_2^2(t_2) & \cdots & \boldsymbol{x}_n^2(t_2) \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \boldsymbol{x}_1^2(t_m) & \boldsymbol{x}_1(t_m)\boldsymbol{x}_2(t_m) & \cdots & \boldsymbol{x}_2^2(t_m) & \cdots & \boldsymbol{x}_n^2(t_m) \end{bmatrix}.$$

One can easily find that the quadratic library has $N = \frac{1}{2}(r^2 + 3r) + 1$ terms. Now we have the sparse optimization problem introduced by standard SINDy algorithm:

$$\boldsymbol{\xi}_j = \underset{\boldsymbol{\xi}'_j}{\operatorname{argmin}} \left\| \dot{\boldsymbol{X}}_{:,j} - \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\xi}'_j \right\|_2^2 + \gamma \|\boldsymbol{\xi}'_j\|_0 \tag{2.17}$$

where $\dot{\boldsymbol{X}}_{:,j}$ denotes the j^{th} column of $\dot{\boldsymbol{X}}$, $\|\cdot\|_0$ counts the number of nonzero elements and γ is a threshold value such that all elements smaller than it are zeroed out, which promotes the sparsity of $\boldsymbol{\Xi}$. One may also notice that $\|\cdot\|_0$ is not convex, which leads to a nonconvex optimization. One way to solve this is instead using ℓ_1 norm resulting in the LASSO, which can be sparsity-promoting as well though it may be computationally expensive [57, 12]. A computationally efficient alternative demonstrated by Brunton *et al.* [7] is to implement the sequential least-squares algorithm which allows a single parameter γ to realize the desired degree of sparsity.

2.2.2 Constrained SINDy

Sparsity can be promoted using sequential least-squares algorithm as mentioned previously, which makes the optimization problem convex as well. And the minimization problem reads,

$$\boldsymbol{\Xi} = \underset{\boldsymbol{\Xi}'}{\operatorname{argmin}} \frac{1}{2} \left\| \dot{\boldsymbol{X}}[:] - \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi}' \right\|_2^2,$$
$$\text{subject to } \boldsymbol{C}\boldsymbol{\xi}' = \boldsymbol{q},$$

where $\boldsymbol{\xi}' = \boldsymbol{\Xi}[:]$ is the vectorized form of the matrix of coefficients. The linear equality constraints $\boldsymbol{C}\boldsymbol{\xi}' = \boldsymbol{q}$ are used to enforce that some entries of $\boldsymbol{\xi}'$ are equal to zero so to achieve sparsity in $\boldsymbol{\Xi}'$ [29]. For example, when the problem is solved iteratively, in each iteration, entries of $\boldsymbol{\xi}$ whose values are smaller than some small value δ_0 are set to zero, where δ_0 can be a constant or the mean of the absolute value of the non-zero entries of $\boldsymbol{\xi}$.

The minimization problem can be also recast as an unconstrained problem,

$$\boldsymbol{\xi} = \underset{\boldsymbol{\xi}', \boldsymbol{\phi}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \boldsymbol{\phi}^T(\boldsymbol{C}\boldsymbol{\xi}' - \boldsymbol{q}) \right] \tag{2.18}$$

where $\hat{\boldsymbol{\Theta}}(\boldsymbol{X}) = \boldsymbol{I} \otimes \boldsymbol{\Theta}(\boldsymbol{X})$, $\dot{\boldsymbol{X}}[:]$ is the vectorized $\dot{\boldsymbol{X}}$ and $\boldsymbol{\phi}$ is a vector representing the Lagrange multiplier.

The linear equality constraints $\boldsymbol{C}\boldsymbol{\xi}' = \boldsymbol{q}$ can also be used to mimic mathematical structures that can be expressed as some linear relationship between the entries of ξ' in a given physical process, rather than just for the purpose of sparsity promotion. An application is that encode the energy-preserving structures observed in fluid and plasma flows [22, 24], which will be introduced in Sec. 3.2.

2.3 Lyapunov stability

Stability plays an important role in system theory. The stability or instability significantly impacts the fundamental characteristics of a system. By analyzing the stability of a system, we gain valuable information about how the system responds to disturbances, how it approaches equilibrium points, and whether it exhibits oscillatory behavior. This qualitative understanding is crucial in various fields of science, engineering, and control theory, as it allows us to predict the system's behavior and design appropriate control strategies to achieve desired outcomes. In an ideal scenario, analytical solutions can directly reveal the long-term behavior of a system. However as mentioned in previous section, generic nonlinear systems such as Eq. (2.1) do not admit closed form solutions. Hence, it is essential to find alternative methods that allow us to determine the stability of the system without the need for analytical solutions of the equations.

The phase portrait as shown in Fig. 2.3, is a powerful tool used in the analysis of dynamical systems to gain qualitative insights into the stability and behavior of systems, particularly concerning fixed points which are also known as equilibrium points. While the phase portrait is often useful, its convenience is limited to simple systems, such as one-dimensional or two-dimensional systems. Unfortunately, it also falls short in providing quantitative information, making it less valuable for future applications. The Russian mathematician and engineer, Lyapunov, introduced a fundamental method for autonomous

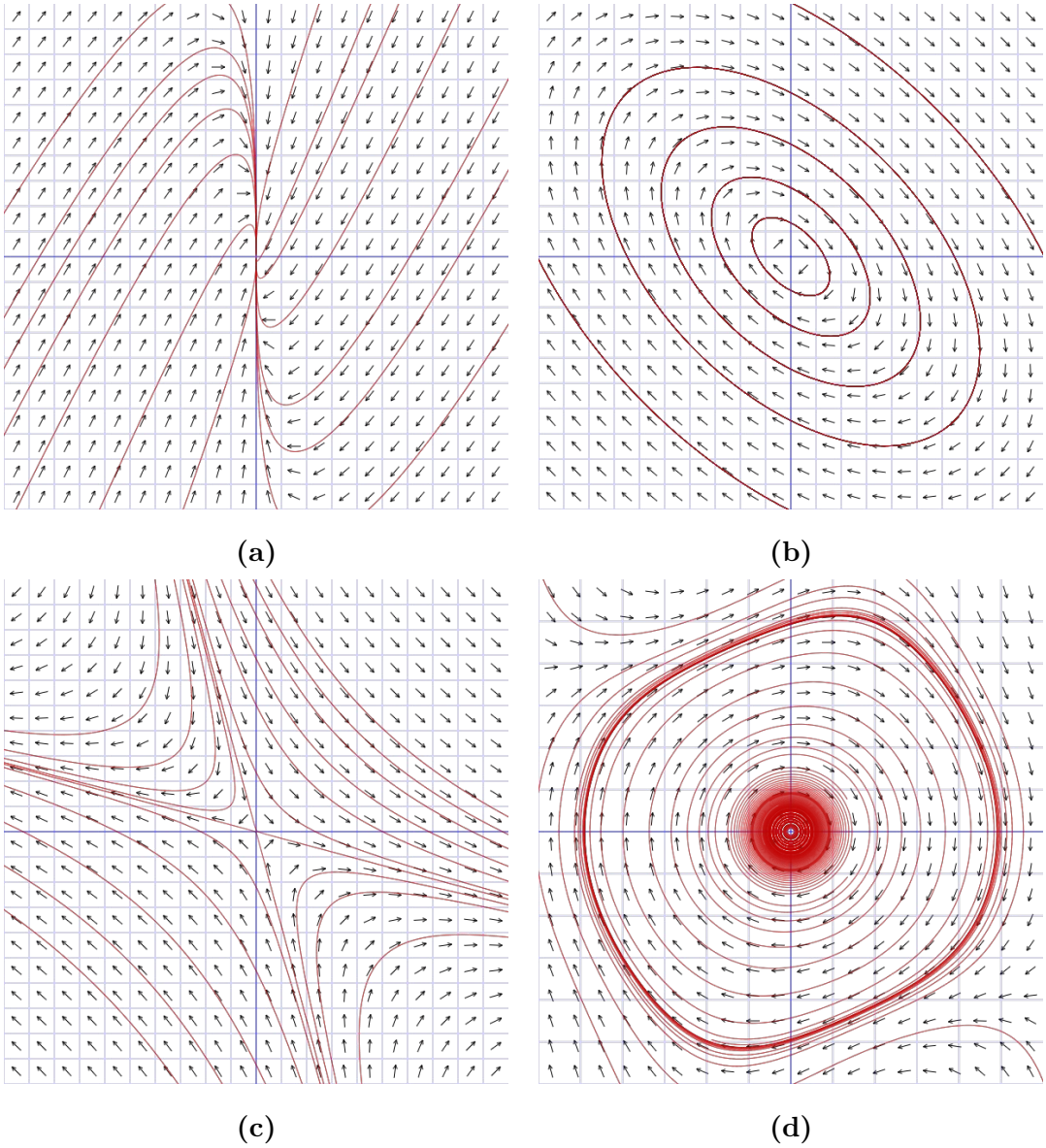

Figure 2.3: Some phase portraits of **(a)** Asymptotically stable **(b)** Marginally stable **(c)** Unstable **(d)** Limit cycle.

systems, now recognized as Lyapunov stability. The Lyapunov stability theorems provide sufficient conditions for various types of stability, including stability, asymptotic stability and more.

Lyapunov's method primarily centers on analyzing the stability of equilibrium points in

dynamical systems. As stability is a central theme in this thesis, we delve into a detailed introduction of this method to aid readers in grasping the essence of our work. Consider a generic autonomous system

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) \tag{2.19}$$

where $\boldsymbol{x} = \boldsymbol{x}(t)$, $\boldsymbol{f} : D \to \mathbb{R}^n$ is a map that is locally Lipschitz continuous from a domain $D \subset \mathbb{R}^n$ into $\mathbb{R}^n$. Without loss of generality, the equilibrium point is at the origin of $\mathbb{R}^n$ for all definitions and theorems we will introduce in this section, which means $\boldsymbol{f}(\mathbf{0}) = \mathbf{0}$. With regard to the equilibrium points, we can define stability as follows:

Definition 2.1. *The equilibrium point $\boldsymbol{x} = \mathbf{0}$ of Eq. (2.19) is*

- *stable, if for every $\varepsilon > 0$, there exists $\delta > 0$ such that*

$$\|\boldsymbol{x}(0)\| < \delta \;\Rightarrow\; \|\boldsymbol{x}(t)\| < \varepsilon,\ \forall\ t \geq 0$$

- *asymptotically stable if it is stable and there exists $\delta > 0$ such that*

$$\|\boldsymbol{x}(0)\| < \delta \;\Rightarrow\; \lim_{t\to\infty} \boldsymbol{x}(t) = 0$$

- *unstable if it is not stable.*

Every dynamical system has trajectories because the evolution of dynamical systems is deterministic. This evolution rule can be defined as a mapping from the phase space to the phase space for each time t:

$$\varphi_t : S \to S$$

so that the position of the system at time t that started at $\boldsymbol{x}_0$ can be denoted as $\boldsymbol{x}(t) = \varphi_t(\boldsymbol{x}_0)$. And if the initial value of time is zero, which means $\varphi_0(\boldsymbol{x}_0) = \boldsymbol{x}_0$, the forward trajectory is the set of subsequent states,

$$\Gamma^+_{\boldsymbol{x}} \equiv \{\phi_t(\boldsymbol{x}) : t \geq 0\}.$$

Trajectories are an essential concept in understanding the behavior of dynamical systems. By analyzing trajectories, we can gain insights into the system's stability, periodic behavior (limit cycles), equilibrium points (fixed points), and other dynamic properties.

Having established some fundamental concepts of describing a system, we can now proceed to introduce the Lyapunov's method. The core idea of the Lyapunov's method is encapsulated in the following theorem [25]:

Theorem 2.1. *For system* (2.19), $\boldsymbol{x} \in D \subset \mathbb{R}^n$ *and the origin is an equilibrium point in the domain* D. *Let* $V : D \to \mathbb{R}$ *be a continuously differentiable function such that*

$$V(\mathbf{0}) = 0 \quad \text{and} \quad V(\boldsymbol{x}) > 0 \quad \text{in} \quad D\backslash\mathbf{0}, \tag{2.20}$$

$$\dot{V}(\boldsymbol{x}) \leq 0 \quad \text{in} \quad D. \tag{2.21}$$

Then, $\boldsymbol{x} = \mathbf{0}$ *is stable. Moreover, if*

$$\dot{V}(\boldsymbol{x}) < 0 \quad \text{in} \quad D\backslash\mathbf{0}, \tag{2.22}$$

then $\boldsymbol{x} = \mathbf{0}$ *is asymptotically stable.*

The continuous differentiable function $V(\boldsymbol{x})$ is called a Lyapunov function if satisfying (2.20) and (2.21). A level surface of a Lyapunov function refers to a surface in the state space where the value of the Lyapunov function is constant, i.e., $V(\boldsymbol{x}) = c$ for some $c > 0$, and this level surface is also called Lyapunov surface. When $\dot{V} \leq 0$, it signifies that as a trajectory crosses a Lyapunov surface $V(\boldsymbol{x}) = c$, it demonstrates an inclination to move within the set $\Omega_c = \{\boldsymbol{x} \in \mathbb{R}^n | V(\boldsymbol{x}) \leq c\}$ and remains confined to this region indefinitely. And when $\dot{V} < 0$, the trajectory moves towards an inner Lyapunov surface with a smaller c from the current Lyapunov surface it is at. So the trajectory will approach to the origin as time progresses until $\dot{V} \geq 0$.

From (2.20) we can see that the strict Lyapunov function $V(\boldsymbol{x})$ is indeed positive definite. That is, for all $\boldsymbol{x} \neq \mathbf{0}$, $V(\boldsymbol{x}) > 0$. Recall that, if we choose a scalar function of quadratic form, we can write,

$$V(\boldsymbol{x}) = \boldsymbol{x}^T \boldsymbol{P} \boldsymbol{x} = P_{ij} x_i x_j,$$

where $\boldsymbol{P}$ is a real symmetric matrix. As V is a positive definite function, it follows that $\boldsymbol{P}$ must be a positive definite matrix which is also called Lyapunov matrix. The Lyapunov matrix can help to determine the stability of a system represented by a linear stability matrix $\boldsymbol{A}$. If the linear stability matrix $\boldsymbol{A}$ is *Hurwitz* which is also called negative definite, then the system exhibits stable behavior in terms of linear stability.

Definition 2.2. *The matrix* $\boldsymbol{A}$ *is called Hurwitz, if all eigenvalues of* $\boldsymbol{A}$ *satisfy* $\mathrm{Re}\lambda_i < 0$*; that is, the real parts of all eigenvalues of A are positive.*

Theorem 2.2. *Consider the Lyapunov matrix equation*

$$\boldsymbol{P}\boldsymbol{A} + \boldsymbol{A}^T\boldsymbol{P} = -\boldsymbol{M}, \tag{2.23}$$

where $\boldsymbol{A}$ *is the linear stability matrix and* $\boldsymbol{M}$ *is a positive definite symmetric matrix. Then,*

- *The matrix* $\boldsymbol{A}$ *is Hurwitz, if and only if for any given* $\boldsymbol{M}$ *there exists a positive definite symmetric matrix* $\boldsymbol{P}$ *that satisfy Eq.* (2.23)*,*
- *if* $\boldsymbol{A}$ *is Hurwitz, then* $\boldsymbol{P}$ *is the unique solution of Eq.* (2.23)*.*

Another related concept for the trajectories of Eq. (2.19) is called "boundedness." Boundedness refers to the property of trajectories remaining within certain limits or bounds over time, without necessarily converging to a specific point or equilibrium. While stability focuses on the convergence of trajectories towards equilibrium points, boundedness considers the overall containment of trajectories within defined boundaries, regardless of their final destinations.

Definition 2.3. *The solutions of Eq.* (2.19) *are bounded if there exists a constant* $\rho > 0$ *that is independent of the initial time* t_0*, and for every* $c \in (0, \rho)$*, there is* $\mu = \mu(c) > 0$ *such that*

$$\|\boldsymbol{x}(t_0)\| \leq c \;\Rightarrow\; \|\boldsymbol{x}(t)\| \leq \mu,\ \forall t \geq t_0.$$

Bounded systems can exhibit trajectories that do not converge to equilibrium points, especially for high-dimensional systems. In high-dimensional systems, the presence of multiple variables and interactions between them can lead to a wide range of dynamic behaviors. Trajectories can exhibit various patterns, including periodic orbits (limit cycles), chaotic behavior, or non-converging behavior. It becomes more challenging to predict the long-term behavior of trajectories as the dimensionality of the system increases. Chaotic systems can indeed exhibit stable behavior within certain regions of their state space, even though long-term prediction of their trajectories may be highly sensitive to initial conditions, making

long-term forecasting essentially impossible. However, within certain regions of the state space, chaotic systems can display stable behavior or have "trapping regions." A trapping region may be defined as in [35].

Definition 2.4. *A set N is a trapping region if it is compact and $\varphi_t(N) \subset \text{int}(N)$ for $t > 0$.*

The $\text{int}(\cdot)$ denotes the interior of a set. So a trapping region is an area in the state space where trajectories remain confined as the system evolves. The existence of trapping regions can be verified or demonstrated using Lyapunov's method [54].

Theorem 2.3. *Consider a Lyapunov function V at the Lyapunov surface $V(\boldsymbol{x}) = c$ in the domain D. V is called strict Lyapunov if $\dot{V} < 0$ for the given c. Denote the set $\{V \leq c_0\}$ by Ω_{V,c_0}. Then Ω_{V,c_0} is a trapping region if it is bounded and V is strict Lyapunov for all $c > c_0$.*

In the upcoming chapter, we will employ these theorems to identify a ball-shaped trapping region for quadratically nonlinear systems. And one of our objectives is to demonstrate the validity of Thm. 2.3 through rigorous analysis and verification of the existence of this trapping region.

Chapter 3

STABILITY ANALYSIS OF QUADRATICALLY NONLINEAR SYSTEMS

Quadratically nonlinear systems are important because of not only mentioned previously that applying commonly used model reduction methods on Navior-Stokes equations result in quadratic models, but also that many nonlinear dynamical systems can be transformed to such form via variable transformations [27, 18, 40, 46]. Meanwhile, the analysis of stability domains is essential for simulations, whether in open or closed-loop scenarios. Lyapunov's direct method is a useful analytical tool to characterize the stability of dynamical systems, in which Lyapunov function plays the core role. A conventional Lyapunov functions is the total kinetic energy, but it may fail the desired purpose, for example, to show stability for interior flows. Indeed, the lack of a systematic approach for finding appropriate Lyapunov function restricts the range of application of Lypapunov's direct method in general. Fortunately, it works well for quadratically nonlinear systems most of the time if the linear stability matrices are normal. In this chapter, we will characterize the stability of quadratically nonlinear systems using Lyapunov's direct method.

3.1 Quadratically nonlinear systems

Consider a quadratically nonlinear system of the form

$$\dot{\boldsymbol{a}} = \boldsymbol{E} + \boldsymbol{L}\boldsymbol{a} + \boldsymbol{Q}\boldsymbol{a}\boldsymbol{a}, \tag{3.1}$$

with state $\boldsymbol{a} = \boldsymbol{a}(t) \in \mathbb{R}^r$, vector $\boldsymbol{E} \in \mathbb{R}^r$, matrix $\boldsymbol{L} \in \mathbb{R}^{r\times r}$ and third-order tensor $\boldsymbol{Q} \in \mathbb{R}^{r\times r\times r}$.

For the convenience of applying Lyapunov stability analysis in 3.4.1, we now introduce a transformation

$$\mathcal{T} : \mathbb{R}^{r\times r\times r} \longrightarrow \mathbb{R}^{r\times r^2},$$

where the operator $\mathcal{T}$ vectorize the last two indices of the third order tensor with dimension $r \times r \times r$ by stacking the columns of the corresponding matrix slices to get a new matrix with the dimension $r \times r^2$. Assume $\boldsymbol{H} = \mathcal{T}(\boldsymbol{Q})$, then the system of Eq. (**??**) can be written as,

$$\dot{\boldsymbol{a}} = \boldsymbol{E} + \boldsymbol{L}\boldsymbol{a} + \boldsymbol{H}(\boldsymbol{a} \otimes \boldsymbol{a}), \tag{3.2}$$

where $\otimes$ denotes the standard Kronecker product. Notice that we previously assume that Q_{ijk} are symmetric in the last two indices, hence $\boldsymbol{H}$ is symmetric in that $\boldsymbol{H}(\boldsymbol{a}_1 \otimes \boldsymbol{a}_2) = \boldsymbol{H}(\boldsymbol{a}_2 \otimes \boldsymbol{a}_1)$ for arbitrary $\boldsymbol{a}_1, \boldsymbol{a}_2 \in \mathbb{R}^r$, which does not change the dynamics [1].

3.2 Flows with energy-preserving nonlinearities

The reason why we are interested in the governing system of (typically partial) differential equations for $\boldsymbol{u}(\boldsymbol{x}, t)$ that are at most quadratic in nonlinearity is that for many fluids, plasmas, and other dynamical systems. Galerkin projection onto the governing equations results in a system of ODEs that takes the exact form of Eq. (3.1). This quadratic form simplifies the analysis and facilitates the application of the Lyapunov's method for stability analysis[48, 54]. The class of systems considered for the original trapping theorem are those with energy-preserving nonlinearity, which is defined as

$$Q_{ijk} a_i a_j a_k = 0, \tag{3.3}$$

or equivalently, for all $i, j, k \in \{1, ..., r\}$,

$$Q_{ijk} + Q_{jik} + Q_{kji} = 0. \tag{3.4}$$

Q_{ijk} are symmetric in swapping j and k without loss of generality. To see why this condition preserves the energy, consider $\overline{\boldsymbol{u}}(\boldsymbol{x})$ being the mean flow and we can introduce the standard kinetic energy K,

$$K = \frac{1}{2}\langle \boldsymbol{u}, \boldsymbol{u} \rangle. \tag{3.5}$$

Then the change of the kinetic energy in time is,

$$\dot{K} = \langle \boldsymbol{u}, \dot{\boldsymbol{u}} \rangle = \langle \overline{\boldsymbol{u}}(\boldsymbol{x}) + a_i(t)\boldsymbol{\chi}_i(\boldsymbol{x}), \dot{a}_j(t)\boldsymbol{\chi}_j(\boldsymbol{x}) \rangle \tag{3.6}$$

$$= (a_i - m_i)\dot{a}_i = (a_i - m_i)\left(E_i + L_{ij}a_j + Q_{ijk}a_j a_k\right), \tag{3.7}$$

$$m_j \equiv -\langle \overline{\boldsymbol{u}}(\boldsymbol{x}), \boldsymbol{\chi}_j(\boldsymbol{x}) \rangle. \tag{3.8}$$

The only cubic term in the temporal modes in Eq. (3.7) is precisely $Q_{ijk}a_i a_j a_k$ and therefore vanishes. The eradication of any cubic terms in $\boldsymbol{a}$, as well as the shift of the energy via $\boldsymbol{m}$ from the presence of a steady-state contribution $\overline{\boldsymbol{u}}(\boldsymbol{x})$, are critical components for the definition of a suitable Lyapunov function, as is illustrated below. Note that this analysis is unchanged if the modes $\boldsymbol{\chi}_j$ are not orthogonal, since this only introduces a mass matrix [42] $M_{ij} = \langle \boldsymbol{\chi}_i, \boldsymbol{\chi}_j \rangle$, which is symmetric by construction and leaves the energy evolution qualitatively unchanged if it is positive definite [26].

3.3 Schlegel and Noack trapping theorem

While energy-preserving systems are bounded as shown in Fig. 3.1, quadratic systems with energy-preserving nonlinearities can be globally bounded as well according to the Schlegel and Noack theorem [48]. The Schlegel and Noack trapping theorem provides necessary and sufficient conditions for energy-preserving, effectively nonlinear, quadratic systems to exhibit a trapping region with upper bound size that can be estimated by $B(\boldsymbol{m}, R_m)$, a ball centered at a vector $\boldsymbol{m} \in \mathbb{R}^r$ with radius R_m. Outside this region the rate of change of energy K is negative everywhere, producing a Lyapunov function that renders this system globally stable. The trapping theorem begins by recentering the origin by a now arbitrary constant vector $\boldsymbol{m}$ so that the energy may be expressed in terms of the shifted state vector $\boldsymbol{y}(t) = \boldsymbol{a}(t) - \boldsymbol{m}$ as

$$K = \frac{1}{2}y_i^2, \tag{3.9}$$

where we use indicial notation here, also referred to as the *Einstein summation convention*, in which a repeated indices have an implied sum. Taking a derivative and substituting in

Eq. (2.14) produces Eq. (3.7), which can be written as

$$\dot{K} = A^S_{ij} y_i y_j + d_i y_i, \tag{3.10}$$

$$A^S_{ij} = L^S_{ij} + (Q_{ijk} + Q_{jik})\, m_k, \quad L^S_{ij} = \frac{1}{2}(L_{ij} + L_{ji}), \quad \text{and} \tag{3.11}$$

$$d_i = E_i + L_{ij} m_j + Q_{ijk} m_i m_j. \tag{3.12}$$

Exploiting the properties that Q_{ijk} are symmetric in the last two indices and energy-preserving, $\boldsymbol{A}^S$ can also be expressed as,

$$A^S_{ij} = L^S_{ij} - Q_{kij} m_k, \quad \text{or} \tag{3.13a}$$

$$\boldsymbol{A}^S = \boldsymbol{L}^S - \boldsymbol{m}^T \boldsymbol{Q}. \tag{3.13b}$$

Then the trapping theorem may now be stated as:

Theorem 3.1. *Consider an effectively quadratically nonlinear system. There exists a monotonically trapping region at least as small as the ball $B(\boldsymbol{m}, R_m)$ if and only if the real, symmetric matrix $\boldsymbol{A}^S$ is negative definite with eigenvalues $\lambda_r \leq \cdots \leq \lambda_1 < 0$; the radius is then given by $R_m = \|\boldsymbol{d}\|_2 / |\lambda_1|$.*[1]

In practice, the goal is then to find an origin $\boldsymbol{m}$ so that the matrix $\boldsymbol{A}^S$ is negative definite, guaranteeing a trapping region and global stability. There is a caveat for systems that do not exhibit effective nonlinearity; effective nonlinearity means that there is no linear subspace that a trajectory starts and remains in indefinitely, so that the quadratic nonlinearities $Q_{ijk} y_j y_k$ never "turn on".

Without effectively quadratic terms, only the backwards direction in Thm 3.1 holds; if we can find an $\boldsymbol{m}$ so that $\boldsymbol{A}^S$ is negative definite, the system exhibits a trapping region. However, such systems can be globally stable without admitting such an $\boldsymbol{m}$. Even when the requirements of the trapping theorem are not fully satisfied, system identification work that incorporates this stability theorem [22] seems to produce models with improved stability properties anyways. The same reference contains an extended discussion of the circumstances under which this theorem holds.

[1] If a system is long-term bounded (not necessarily exhibiting a monotonically trapping region) and effectively nonlinear, the existence of an $\boldsymbol{m}$ producing a negative semidefinite $\boldsymbol{A}^S$ is still guaranteed.

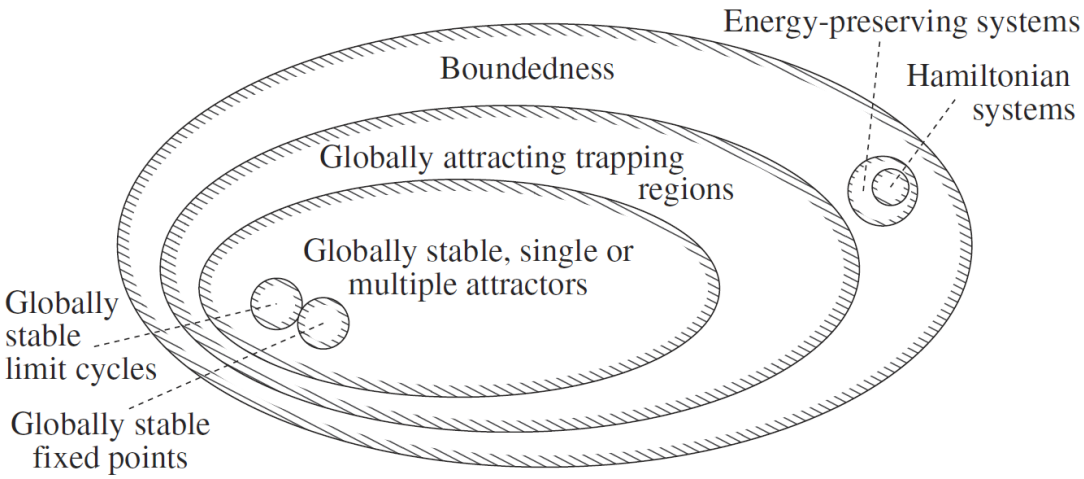

Figure 3.1: Venn Diagram of dynamical systems from Schlegel and Noack [48].

3.4 Analytical estimates of the local stability of quadratically nonlinear systems

In this section, we exploit the Lyapunov's direct method to explore the stability of a quadratically nonlinear system of the form of Eq. (3.1). We again look into the energy-preserving structure exhibited by quadratic nonlinearities in the system, however slightly break the structure so that the system is not strictly energy-preserving in these terms. In this way we examine and discuss the stability of the system. Two important quantities derived in this section will help us identify and approximate the stability of the dynamic systems.

3.4.1 Stability domain of quadratically nonlinear systems

Instead of providing a tight upper bound of trapping region size of the energy-preserving quadratic systems [48], we consider the situation where energy-preserving nonlinearities no longer exists in the quadratically nonlinear systems. However, we assume that the sums of the quadratic coefficients over index cyclic permutations are bounded by a small amount

ϵ_Q, i.e.,

$$-\epsilon_Q \leq Q_{ijk} + Q_{jik} + Q_{kji} \leq \epsilon_Q, \tag{3.14}$$

which means that it slightly breaks the energy-preserving property of the quadratic term of the dynamical system. Hence, the cubic term in $\dot{K}$ does not vanish in this case. Based on this constraint, consider the quadratically nonlinear system of the form (3.1), and for generalization of Lyapunov's direct method, we introduce an arbitrary state $\boldsymbol{m}$ as the new shifted origin, so that the dynamical system expressed in this new translated coordinates $\boldsymbol{y} = \boldsymbol{a} - \boldsymbol{m}$ is,

$$\dot{\boldsymbol{y}} = \boldsymbol{d} + \boldsymbol{A}\boldsymbol{y} + \boldsymbol{Q}\boldsymbol{y}\boldsymbol{y}, \tag{3.15}$$

or

$$\dot{\boldsymbol{y}} = \boldsymbol{d} + \boldsymbol{A}\boldsymbol{y} + \boldsymbol{H}(\boldsymbol{y} \otimes \boldsymbol{y}), \tag{3.16}$$

with $\boldsymbol{d}$ defined in Eq. (3.12), $\boldsymbol{H} = \mathcal{T}(\boldsymbol{Q})$ and

$$A_{ij} = L_{ij} + (Q_{ijk} + Q_{ikj})m_k. \tag{3.17}$$

Then we define a Lyapunov function $v(\boldsymbol{y})$ given a Lyapunov matrix $\boldsymbol{P}$,

$$v(\boldsymbol{y}) = \frac{1}{2}\langle \boldsymbol{y}, \boldsymbol{y} \rangle_{\boldsymbol{P}} \tag{3.18a}$$

$$= \frac{1}{2}\boldsymbol{y}^T \boldsymbol{P}\boldsymbol{y}, \tag{3.18b}$$

with the time derivative of $v(\boldsymbol{y})$

$$\dot{v}(\boldsymbol{y}) = \frac{1}{2}\left(\langle \boldsymbol{y}, \dot{\boldsymbol{y}} \rangle_{\boldsymbol{P}} + \langle \dot{\boldsymbol{y}}, \boldsymbol{y} \rangle_{\boldsymbol{P}}\right) \tag{3.19a}$$

$$= \boldsymbol{y}^T \boldsymbol{P}\dot{\boldsymbol{y}}. \tag{3.19b}$$

Since the domain of boundedness $\boldsymbol{\Omega_m}$ can be complicated geometrically, we consider closed balls $\mathbf{B} \subseteq \boldsymbol{\Omega_m}$ as the candidates of regions where the solutions of Eq. (3.15) are bounded,

$$\mathbf{B}_{\boldsymbol{y}}(\boldsymbol{R}) = \{\boldsymbol{y} \in \mathbb{R}^r \mid \|\boldsymbol{y}\| \leq \boldsymbol{R}\}.$$

We now lower-bound the domain of boundedness by the set $\mathbf{B}_{\boldsymbol{y}}$. The task then becomes to make this bound as tight as possible, which is, to find $\sup(\boldsymbol{R})$ as the estimate of the local stability radius. This leads to the following theorem.

Theorem 3.2. *Regarding the system* (3.15) *and a Lyapunov matrix* $\boldsymbol{P}$*, there exists a trapping region if the following statements are true:*

a. All eigenvalues of $\tilde{\boldsymbol{A}}^S = \boldsymbol{P}\boldsymbol{A}^S$*, where* $\boldsymbol{A}^S$ *is defined by* (3.11) *are negative, i.e.* $\lambda_i(\tilde{\boldsymbol{A}}^S) < 0, i = 1, \cdots, r$.

b. $\lambda_1^2(\tilde{\boldsymbol{A}}^S) - \frac{4r^{\frac{3}{2}}\epsilon_Q}{3}\|\tilde{\boldsymbol{d}}\|_2 \geq 0$*, where* $\epsilon_Q = \|Q_{ijk} + Q_{ijk} + Q_{kji}\|_{max}$ *and* $\tilde{\boldsymbol{d}} = \boldsymbol{P}\boldsymbol{d}$.

And the smallest radius of the trapping region $\mathbf{B}_{\boldsymbol{y}}(\rho_-)$ *is given by*

$$\rho_- = -\frac{3}{2r^{\frac{3}{2}}\epsilon_Q}\left(\sqrt{\lambda_1^2(\tilde{\boldsymbol{A}}^S) - \frac{4r^{\frac{3}{2}}\epsilon_Q}{3}\|\tilde{\boldsymbol{d}}\|_2} + \lambda_1(\tilde{\boldsymbol{A}}^S)\right),$$

while all trajectories start inside $\mathbf{B}_{\boldsymbol{y}}(\rho_+)$ *are bounded where*

$$\rho_+ = \frac{3}{2r^{\frac{3}{2}}\epsilon_Q}\left(\sqrt{\lambda_1^2(\tilde{\boldsymbol{A}}^S) - \frac{4r^{\frac{3}{2}}\epsilon_Q}{3}\|\tilde{\boldsymbol{d}}\|_2} - \lambda_1(\tilde{\boldsymbol{A}}^S)\right).$$

Proof. The time derivative of the Lyapunov matrix given by Eq. (3.19) is

$$\dot{v}(\boldsymbol{y}) = \boldsymbol{y}^T\boldsymbol{P}\dot{\boldsymbol{y}} \tag{3.20a}$$

$$= \boldsymbol{d}^T\boldsymbol{P}\boldsymbol{y} + \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{A}^S\boldsymbol{y} + \boldsymbol{y}^T\boldsymbol{P}\boldsymbol{Q}\boldsymbol{y}\boldsymbol{y} \tag{3.20b}$$

$$= \tilde{\boldsymbol{d}}^T\boldsymbol{y} + \boldsymbol{y}^T\tilde{\boldsymbol{A}}^S\boldsymbol{y} + \boldsymbol{y}^T\tilde{\boldsymbol{Q}}\boldsymbol{y}\boldsymbol{y}, \tag{3.20c}$$

where $\tilde{\boldsymbol{d}} = \boldsymbol{P}\boldsymbol{d}$, $\tilde{\boldsymbol{A}}^S = \boldsymbol{P}\boldsymbol{A}^S$ and $\tilde{\boldsymbol{Q}} = \boldsymbol{P}\boldsymbol{Q}$. We can then introduce $\tilde{Q}'_{ijk} = \tilde{Q}_{ijk} + \tilde{Q}_{jki} + \tilde{Q}_{kij}$, and $\tilde{\boldsymbol{H}}_0 = \mathcal{T}(\tilde{\boldsymbol{Q}}')$ so that $\dot{v}(\boldsymbol{y})$ now is

$$\dot{v}(\boldsymbol{y}) = \tilde{\boldsymbol{d}}^T\boldsymbol{y} + \boldsymbol{y}^T\tilde{\boldsymbol{A}}^S\boldsymbol{y} + \frac{1}{3}\boldsymbol{y}^T\tilde{\boldsymbol{H}}_0(\boldsymbol{y}\otimes\boldsymbol{y}) \tag{3.21}$$

$$\leq \|\tilde{\boldsymbol{d}}^T\boldsymbol{y}\|_2 + \lambda_{\max}(\tilde{\boldsymbol{A}}^S)\|\boldsymbol{y}\|_2^2 + \frac{1}{3}\|\tilde{\boldsymbol{H}}_0\|_2\|\boldsymbol{y}\|_2^3 \tag{3.22}$$

$$\leq \|\tilde{\boldsymbol{d}}\|_2\|\boldsymbol{y}\|_2 + \lambda_1(\tilde{\boldsymbol{A}}^S)\|\boldsymbol{y}\|_2^2 + \frac{r^{\frac{3}{2}}\epsilon_Q}{3}\|\boldsymbol{y}\|_2^3. \tag{3.23}$$

For $\|\tilde{\boldsymbol{d}}\|_2\|\boldsymbol{y}\|_2 + \lambda_1(\tilde{\boldsymbol{A}}^S)\|\boldsymbol{y}\|_2^2 + \frac{r^{\frac{3}{2}}\epsilon_Q}{3}\|\boldsymbol{y}\|_2^3 < 0$, we must have

$$\rho_- < \|\boldsymbol{y}\|_2 < \rho_+$$

where

$$\rho_+ = \frac{3}{2r^{\frac{3}{2}}\epsilon_Q}\left(\sqrt{\lambda_1^2(\tilde{\boldsymbol{A}}^S) - \frac{4r^{\frac{3}{2}}\epsilon_Q}{3}\|\tilde{\boldsymbol{d}}\|_2} - \lambda_1(\tilde{\boldsymbol{A}}^S)\right), \tag{3.24}$$

$$\rho_- = -\frac{3}{2r^{\frac{3}{2}}\epsilon_Q}\left(\sqrt{\lambda_1^2(\tilde{\boldsymbol{A}}^S) - \frac{4r^{\frac{3}{2}}\epsilon_Q}{3}\|\tilde{\boldsymbol{d}}\|_2} + \lambda_1(\tilde{\boldsymbol{A}}^S)\right). \tag{3.25}$$

And to guarantee ρ_+ and ρ_- both being real, $\Delta = \lambda_1^2(\tilde{\boldsymbol{A}}^S) - \frac{4r^{\frac{3}{2}}\epsilon_Q}{3}\|\tilde{\boldsymbol{d}}\|_2$ must be non-negative. For all $\boldsymbol{y} \in (\rho_-, \rho_+)$, we now have $v(\boldsymbol{y}) > 0$ and $\dot{v}(\boldsymbol{y}) < 0$, which yields the claimed result. □

3.4.2 Interpretation of the local stability radius

The ρ_+ and ρ_- are the two nontrivial roots of Eq. (3.23) and the different dynamical regions of state space are illustrated in Fig. 3.2. ρ_+ is a conservative estimate of the diameter of the inscribed ball $\mathbf{B}_{\rho_+}$ in the locally stable region $\boldsymbol{\Omega}_{\rho_+}$. Meanwhile ρ_- is an estimate of the diameter of the set Ω_{ρ_-} and $\dot{v} = 0$ at the boundary $\partial\Omega_{\rho_-}$. Negative energy growth in $\mathbf{B}_{\rho_+} \setminus \mathbf{B}_{\rho_-}$ is guaranteed according to Thm. 3.2 and inside $\mathbf{B}_{\rho_-}$ there exists a region $\boldsymbol{\Omega}_{\rho_-}$ inside which positive energy growth is expected to be observed. Then the long-time behavior of solutions to the system (3.16) with initial conditions $\boldsymbol{y}_0 \in \mathbf{B}_{\rho_+}$ alternates between $\mathbf{B}_{\rho_-} \setminus \boldsymbol{\Omega}_{\rho_-}$ and $\boldsymbol{\Omega}_{\rho_-}$. In a special case where $\boldsymbol{y}_0$ is a real solution to Eq. (3.21), the long-time behavior of the system stays at the boundary of $\boldsymbol{\Omega}_{\rho_-}$. And it is not difficult to get that

$$\lim_{\epsilon_Q \to 0} \rho_+ = \infty, \tag{3.26a}$$

$$\lim_{\epsilon_Q \to 0} \rho_- = -\frac{\|\tilde{\boldsymbol{d}}\|_2}{\lambda_1(\tilde{\boldsymbol{A}}^S)}, \tag{3.26b}$$

which correctly reproduces the original trapping theorem guaranteeing global boundedness (in fact, a monotonically trapping region). We can also validate the limit by setting ϵ_Q to

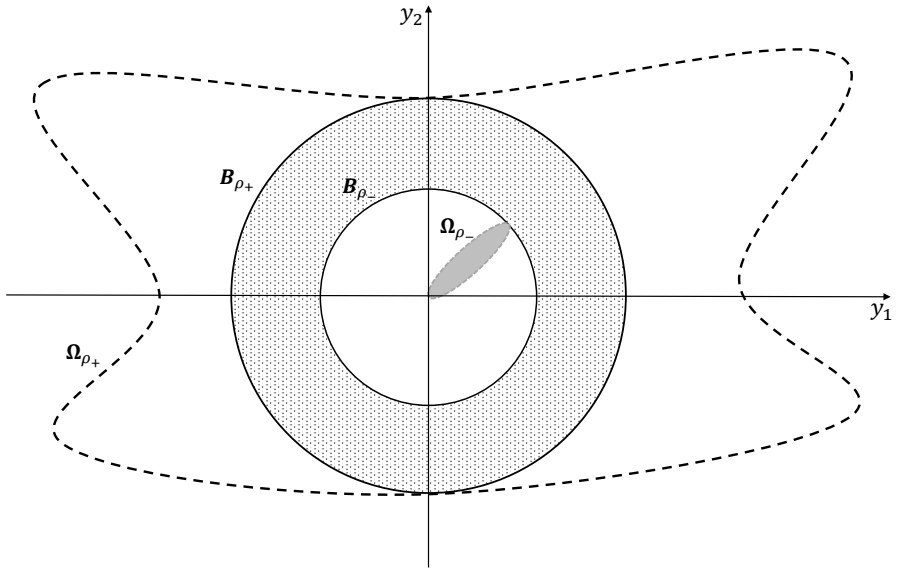

Figure 3.2: Sketch of the trapping region. $\dot{v} < 0$ is observed in the area between the dashed line and outside the grey area $\boldsymbol{\Omega}_{\rho_-}$. $\boldsymbol{\Omega}_{\rho_+}$ has a complex hyperbolic geometry and $\boldsymbol{\Omega}_{\rho_-}$ has a shape close to ellipsoid, which is because the third-order terms are usually very small near the origin so it has little contribution to the magnitude of v there. The local stability is guaranteed in the ball-shape region $\boldsymbol{B}_{\rho_-}$.

zero so that $\tilde{\boldsymbol{H}}_0 = 0$, then the time derivative of the Lyapunov function (3.21) becomes

$$\begin{aligned}\dot{v}(\boldsymbol{y}) &= \tilde{\boldsymbol{d}}^T\boldsymbol{y} + \boldsymbol{y}^T\tilde{\boldsymbol{A}}^S\boldsymbol{y} \\ &\leq \|\tilde{\boldsymbol{d}}\|_2\|\boldsymbol{y}\|_2 + \lambda_1(\tilde{\boldsymbol{A}}^S)\|\boldsymbol{y}\|_2^2 \\ &= \|\boldsymbol{y}\|_2\left(\|\tilde{\boldsymbol{d}}\|_2 + \lambda_1(\tilde{\boldsymbol{A}}^S)\|\boldsymbol{y}\|_2\right),\end{aligned}$$

from which we can see that for all $\|\boldsymbol{y}\|_2 > -\frac{\|\tilde{\boldsymbol{d}}\|_2}{\lambda_1(\tilde{\boldsymbol{A}}^S)}$, $\dot{v} < 0$. And an estimate of the radius of the ball-shape trapping region is given by $R_{\boldsymbol{m}} = -\frac{\|\tilde{\boldsymbol{d}}\|_2}{\lambda_1(\tilde{\boldsymbol{A}}^S)}$.

Note that the "shell" of stability vanishes when $\rho_+ = \rho_-$ or equivalently when the square root vanishes:

$$\epsilon_Q = \frac{3\lambda_1^2(\tilde{\boldsymbol{A}}^S)}{4r^{\frac{3}{2}}\|\tilde{\boldsymbol{d}}\|_2}, \tag{3.27}$$

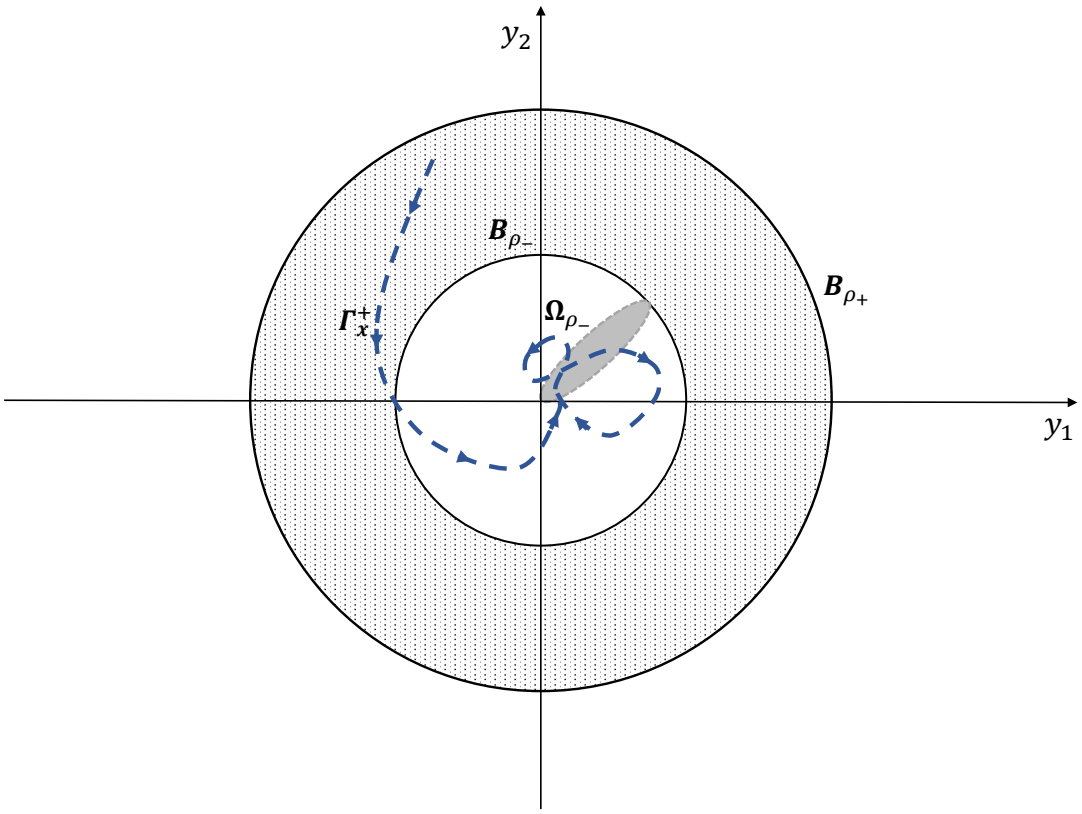

Figure 3.3: An example trajectory (dashed blue) beginning and remaining in the trapping region.

which provides a condition on the smallness of the totally symmetric part of the quadratic nonlinearity in order for this shell of stability to exist. Note that this condition becomes more stringent in higher dimensional state spaces, although there are also more parameters in $\boldsymbol{m}$ that may be able to make $\|\boldsymbol{d}\|_2$ smaller (the region near the origin gets more stable) or $\lambda_1^2(\boldsymbol{A}^S)$ bigger (the unstable region where the non-energy-preserving quadratic nonlinearities begin to dominate gets shifted outwards). Specially, when $\|\boldsymbol{d}\|_2 = 0$, $\boldsymbol{\Omega}_{\rho_-}$ collapses to an asymptotically stable fixed point $\boldsymbol{m}$ and $\mathbf{B}_{\rho_-}$ vanishes.

Chapter 4

PROMOTING LOCAL STABILITY IN DATA-DRIVEN MODELS OF QUADRATICALLY NONLINEAR DYNAMICS

We now have a theorem for promoting trapping regions that mean trajectories starting in B_{ρ_+} will remain in this region, we can promote this stability when we find models directly from data. This can be done with any optimization based method such as the system identification of nonlinear dynamical systems (SINDy) [7].

4.1 Extended trapping SINDy algorithm

4.1.1 Trapping SINDy algorithm

Consider the dynamical system (3.16) and the original SINDy algorithm with l_1 regularizer to identify such system,

$$\boldsymbol{\Xi} = \underset{\boldsymbol{\Xi}'}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}} - \boldsymbol{\Theta}(\boldsymbol{X})\boldsymbol{\Xi}'\|_2^2 + \gamma \|\boldsymbol{\Xi}'\|_1 \right],$$

where $\boldsymbol{X} \in \mathbb{R}^{r \times M}$ is the collection of snapshots of the system, consisting data from a low-dimensional basis like the POD, i.e. $\boldsymbol{X} = [\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_r]$, candidate functions library $\boldsymbol{\Theta} \in \mathbb{R}^{r \times N}$ contains terms up to quadratic polynomials in $\boldsymbol{a}$ and $\boldsymbol{\Xi} \in \mathbb{R}^{N \times r}$ is the coefficients matrix.

To encode the enforced energy-preserving constraint on the quadratics terms, Loiseau *et al.* [29] raised the constrained SINDy using vectorized $\boldsymbol{\Xi}[:] = \boldsymbol{\xi} \in \mathbb{R}^{rN}$ to recast the constrained minimization problem as an unconstrained convex problem using an augmented functional formulation. As introduced previously, the constraint is encoded as $\boldsymbol{C}\boldsymbol{\xi} = \boldsymbol{q}$, so that the optimization problem reads,

$$\boldsymbol{\xi} = \underset{\boldsymbol{\xi}', \boldsymbol{\phi}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \gamma \|\boldsymbol{\xi}'\|_1 + \boldsymbol{\phi}^T(\boldsymbol{C}\boldsymbol{\xi}' - \boldsymbol{q}) \right], \tag{4.1}$$

where $\hat{\boldsymbol{\Theta}}(\boldsymbol{X}) = \boldsymbol{I} \otimes \boldsymbol{\Theta}(\boldsymbol{X})$, $\dot{\boldsymbol{X}}[:]$ is the vectorized $\dot{\boldsymbol{X}}$ so that we have $\hat{\boldsymbol{\Theta}} \in \mathbb{R}^{rM \times rN}$, $\dot{\boldsymbol{X}}[:] \in \mathbb{R}^{rM}$. The number of constraints of the energy-preserving structure is given by $p =$

$r(r+1)(r+2)/6$ and therefore we have $\boldsymbol{C} \in \mathbb{R}^{p \times rN}$, $\boldsymbol{q} \in \mathbb{R}^p$ and the Lagrange multiplier $\boldsymbol{\phi} \in \mathbb{R}^p$. Additionally, for the linear equality constraints of the energy-preserving structure, $\boldsymbol{q} = 0$ and the number of free parameters is given by $rN - p = 2p$ [22]. Furthermore, these constraints can be used to impose any linear relationship that describe a physical process or enforce *a priori* known values of given entries ξ_i as well.

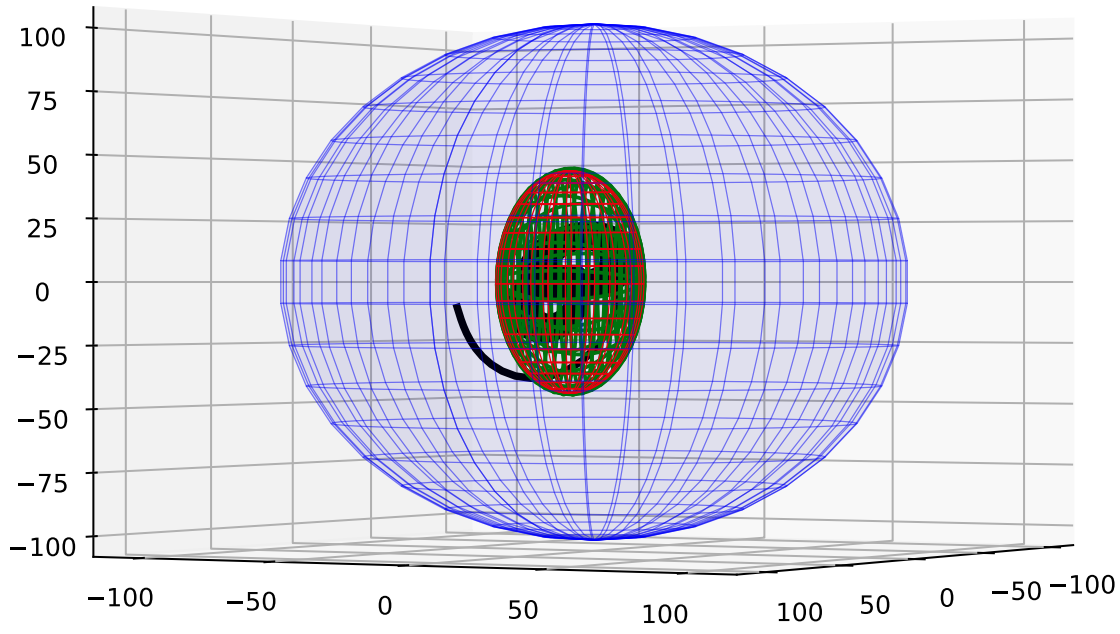

Figure 4.1: Illustration of trapping SINDy model of Lorenz system [22]. The black solid line is a trajectory from trapping SINDy model prediction. Positive energy growth can be observed in the red ellipsoid given by the algorithm. The green ellipsoid is given by analytic solution to the Lorenz. And the blue ball-shape trapping region is estimated by the ellipsoid of positive energy growth.

It is still not enough to guarantee the global stability of the linear-quadratic systems with the energy-preserving constraint. Based on Schlegel and Noack trapping theorem, Kaptanoglu *et al.* [22] enforce the global stability of the quadratically nonlinear dynamics by searching for an $\boldsymbol{m}$ that makes $\tilde{\boldsymbol{A}}^S$ *Hurwitz* without breaking the energy-preserving structure of the systems. In other words, the largest real part of the eigenvalue of $\tilde{\boldsymbol{A}}^S$

should be negative, then the optimization problem becomes,

$$\boldsymbol{\xi} = \underset{\boldsymbol{\xi}', \boldsymbol{\phi}, \boldsymbol{m}, \boldsymbol{A}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \gamma \|\boldsymbol{\xi}'\|_1 + \boldsymbol{\phi}^T(\boldsymbol{C}\boldsymbol{\xi}' - \boldsymbol{q}) + \frac{\boldsymbol{\lambda}_1(\tilde{\boldsymbol{A}}^S)}{\eta} \right], \tag{4.2}$$

where the new term denotes that the largest eigenvalue of $\tilde{\boldsymbol{A}}^S$ will be enforced to be negative so that $\tilde{\boldsymbol{A}}^S$ is *Hurwitz*, and the magnitude of whose loss is modulated by the hyperparameter η. The strategy of enforcing the $\tilde{\boldsymbol{A}}^S$ to be *Hurwitz* is to treat $\boldsymbol{\xi}$ and $\boldsymbol{m}$ as independent, which is, to solve the convex problem in $\boldsymbol{\xi}$ and use the optimized $\boldsymbol{\xi}^*$ to solve for $\boldsymbol{m}$. At each iteration we repeat this procedure until tolerance or maximum number of iteration is reached. Note that the convergence properties remain unclear, however this algorithm shows fairly good performance. Then a relax-and-split approach is adopted to produce the algorithm to obtain the *Hurwitz* matrix $\boldsymbol{A}^S$ [61]. To be specific, the coefficients in $\boldsymbol{\xi}$ are related to the matrix $\boldsymbol{A}^S$ by an operator $\boldsymbol{P}^A$ which is formulated by two projection operators $\boldsymbol{P}^L \in \mathbb{R}^{r\times r\times rN}$ and $\boldsymbol{P}^Q \in \mathbb{R}^{r\times r\times r\times rN}$,

$$\begin{aligned} L_{ij}^S &= P_{ijk}^L \xi_k, \\ Q_{ijk} &= P_{ijkl}^Q \xi_l. \end{aligned}$$

Then we have,

$$\begin{aligned} \boldsymbol{A}_{ij}^S &= P_{ijl}^L \xi_l + \left(P_{ijkl}^Q + P_{jikl}^Q \right) \xi_l m_k \\ &= \left(P_{ijl}^L + \left(P_{ijkl}^Q + P_{jikl}^Q \right) m_k \right) \xi_l. \end{aligned}$$

Therefore, the operator $\boldsymbol{P}^A \in \mathbb{R}^{r\times r\times rN}$ is given by $P_{ijl}^A = P_{ijl}^L + \left(P_{ijkl}^Q + P_{jikl}^Q \right) m_k$. An auxiliary variable $\boldsymbol{A}$ is then introduced to represent the projection $\boldsymbol{A}^S = \boldsymbol{P}^A\boldsymbol{\xi}$ on to the space of negative definite matrices. By minimizing the difference between $\boldsymbol{A}^S$ and $\boldsymbol{A}$, $\boldsymbol{A}^S$ is enforced to be negative definite.

4.1.2 Extended trapping SINDy algorithm

Ideally, we would expect linear-quadratic systems with the quadratic energy-preserving structure to be globally stable and exhibit a monotonically trapping region according to Schlegel and Noack theorem. However, for quadratically nonlinear dynamics that do not

have this energy-preserving structure, the stability is not guaranteed and generic quadratically nonlinear systems are highly possible to be globally unbounded. Hence it is desirable to promote the local Lyapunov stability of the origin with some variations of the original trapping algorithm.

Similarly to Sawant *et al.* [47], one can also guarantee local stability and promote large stable radii for data-driven models with nonlinearities that do not preserve the energy by minimizing the destabilizing effects of the cubic terms in $\dot{v}(\boldsymbol{y})$. The optimization reads,

$$\boldsymbol{\xi} = \underset{\boldsymbol{\xi}', \boldsymbol{m}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \gamma \|\boldsymbol{\xi}'\|_1 + \alpha^{-1} \|\boldsymbol{H}\|_F^2 \right], \tag{4.3}$$

where the matrix $\boldsymbol{H}$ can be related to the model coefficients by a projection operators. And the regularizer is convex then the optimization problem is convex. Sawant *et al.* leverage this advantage and make the matrix $\boldsymbol{A}^S$ *Hurwitz* via a post-processing step hence the optimization problem is much easier to solve. However, this a-posteriori fixing of the *Hurwitz* condition for $\boldsymbol{A}^S$ has the downside that it may take the solution significantly out of the local minimum found during optimization. For this reason, we combine the two loss functions in Eq. (4.2) and (4.3) without the energy-preserving constraint to get,

$$\boldsymbol{\xi} = \underset{\boldsymbol{\xi}', \boldsymbol{\phi}, \boldsymbol{m}, \boldsymbol{A}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \gamma \|\boldsymbol{\xi}'\|_1 + \eta^{-1} \boldsymbol{\lambda}_1(\tilde{\boldsymbol{A}}^S) + \alpha^{-1} \|\boldsymbol{H}\|_F^2 \right]. \tag{4.4}$$

This loss function promotes a *Hurwitz* $\boldsymbol{A}^S$ matrix while minimizing the destabilizing effects of the cubic contributions in $\dot{v}(\boldsymbol{y})$. And an estimate of stability radius of this method is given by Kramer [26]. It is important to keep in mind that the last loss term in Eq. (4.3) is only motivated in the non-energy-preserving nonlinear setting. Moreover, $\boldsymbol{A}^S$ is defined by Eq. (3.11) instead of Eq. (3.13) in this setting since the energy-preserving constraints do not exist here. In principle, one could instead reformulate the last loss term as an inequality constraint, but we refrain from this strategy because it would involve inequality constraints that are not affine in the model coefficients.

We now additionally promote local stable models without minimizing the nonlinear effect of the systems. Recall from Thm. 3.2 that under the certain condition, linear-quadratic models can be locally bounded and exhibit a trapping region even if the hard constraint of the energy-preserving structure is relaxed. For ease of notation, we introduce $Q'_{ijk} =$

Table 4.1: Description of extended trapping SINDy hyperparameters.

γ	Specifies the strength of sparsity-promotion through the l_1 regularizer. For all experiment results in this thesis, $\gamma = 0$.
η	Specifies how strongly to push the algorithm towards models with *Hurwitz* $\tilde{\boldsymbol{A}}^S$.
β	Determines how large the deviation is from $\boldsymbol{Q}$ having zero totally symmetric part.

$Q_{ijk} + Q_{jki} + Q_{kij}$ and $\boldsymbol{H}_0 = \mathcal{T}(\boldsymbol{Q}')$. The objective function now reads,

$$\underset{\boldsymbol{\xi}', \boldsymbol{m}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \gamma \|\boldsymbol{\xi}'\|_1 + \eta^{-1} \boldsymbol{\lambda}_1(\tilde{\boldsymbol{A}}^S) \right] \quad s.t. \quad \|\boldsymbol{H}_0\|_{max} = \epsilon_Q, \tag{4.5}$$

or the form of unconstrained optimization,

$$\underset{\boldsymbol{\xi}', \boldsymbol{m}}{\operatorname{argmin}} \left[\frac{1}{2} \|\dot{\boldsymbol{X}}[:] - \hat{\boldsymbol{\Theta}}(\boldsymbol{X})\boldsymbol{\xi}'\|_2^2 + \gamma \|\boldsymbol{\xi}'\|_1 + \eta^{-1} \boldsymbol{\lambda}_1(\tilde{\boldsymbol{A}}^S) + \beta^{-1} \|\boldsymbol{H}_0\|_F^2 \right]. \tag{4.6}$$

Eq. (4.6) is quite similar to the objective function used in Ouala *et al.* [39]. This optimization can be understood as maximizing the stability radius about the origin by promoting energy-preserving structure in $\boldsymbol{Q}$, instead of maximizing the stability radius about the radius by minimizing the norm of $\boldsymbol{Q}$ as in Sawant *et al.* [47] to make the nonlinearities weak. Considering that there is no truly "zero" due to the existence of machine precision in computer arithmetic, even the resulting models from Eq. (4.2) will not be truly globally bounded numerically. Hence the local stability radius from Thm. 3.2 could be referenceable for data-driven models, especially for those with weakly energy-preserving quadratic nonlinearities.

Notice that we still need $\tilde{\boldsymbol{A}}^S$ to be *Hurwitz* to get the estimated stability radius, and the largest eigenvalue λ_{max} of $\tilde{\boldsymbol{A}}^S$ decides the stability radius directly. So we might be able to promote the local stability by choosing a different Lyapunov matrix $\boldsymbol{P}$ to control the magnitude of λ_{max} as well.

Although the optimization settings of Eq. (4.4) and Eq. (4.6) seem to be similar and can both promote models that are locally stable by construction, the intentions are noticeably different. For Eq. (4.4), apart from getting stable data-driven models with large stable radii, this approach also provides a way to examine the linear stability by punishing the last term hard to get linearized models about the origins. However one may notice that we could also

lose much information of the nonlinearities of the physical systems using this optimization, which is marginally helpful for ones who are interested in the nonlinear effects of the cubic terms in $\dot{v}(\boldsymbol{y})$ or the behavior of the true models at very large distance of the origin. In other words, this method of directly punishing the strength of the nonlinearities requires that the nonlinearities of the systems to be weak so that the data-driven models can perform well in identification tasks.

Instead of minimizing the $\|\boldsymbol{H}\|_F$, Eq. (4.6) minimizes $\|\boldsymbol{H}_0\|_F$, i.e. only the deviation of the nonlinearities from the energy-preserving structure. After all, this is the only part of the nonlinearities that contributes to the energy evolution. To explain further, we do not promote the local stability of the models by weakening the nonlinearities of the system, but by enforcing a weakly quadratic energy-preserving structure. Moreover, we here use this optimization to find the ρ_+ and ρ_-, which guarantee the stability of the model based on Thm. 3.2. Specifically, the size of the domain of boundedness ρ_+ and the size of the trapping region ρ_- is derived from the optimized $\|H_0\|_{max}$. On the other hand, the magnitude of ρ_+ provides a quantity that can evaluate the performance of the optimization in terms of stability as well, which Ouala *et al.* did not have available for further analyzing their model stability.

The globally stable results from Kaptanoglu *et al.* [22] can be additionally reproduced through Eq. 4.6 by sufficient penalization of the maximal deviation from energy-preserving nonlinear structure. As what we show in Sec. 3.4, global stability of the models can be achieved by reducing the loss of $\|\boldsymbol{H}_0\|$ to machine precision.

4.2 Results

We now evaluate the performance of our extended trapping SINDy algorithm on tasks and capabilities including model identification, simulation and noise robustness for a few canonical systems which are commonly seen in fluid dynamics.Two widely used and appropriate definitions to assess the model quality are the time-average error E_f in $\dot{\boldsymbol{X}}$ and for models

with closed forms, the relative Frobenius error E_{coef} in $\boldsymbol{\Xi}$,

$$E_f = \frac{\overline{\|\dot{\boldsymbol{X}}_{\text{True}} - \dot{\boldsymbol{X}}_{\text{SINDy}}\|^2}}{\|\dot{\boldsymbol{X}}_{\text{True}}\|^2}, \tag{4.7a}$$

$$E_{coef} = \frac{\|\boldsymbol{\Xi}_{\text{True}} - \boldsymbol{\Xi}_{\text{SINDy}}\|_F}{\|\boldsymbol{\Xi}_{\text{True}}\|_F}. \tag{4.7b}$$

And a commonly seen quantity to evaluate the performance of data-driven models is the relative prediction error,

$$E_{pred} = \frac{\|\boldsymbol{X}_{\text{True}} - \boldsymbol{X}_{\text{SINDy}}\|_F}{\|\boldsymbol{X}_{\text{True}}\|_F}. \tag{4.8}$$

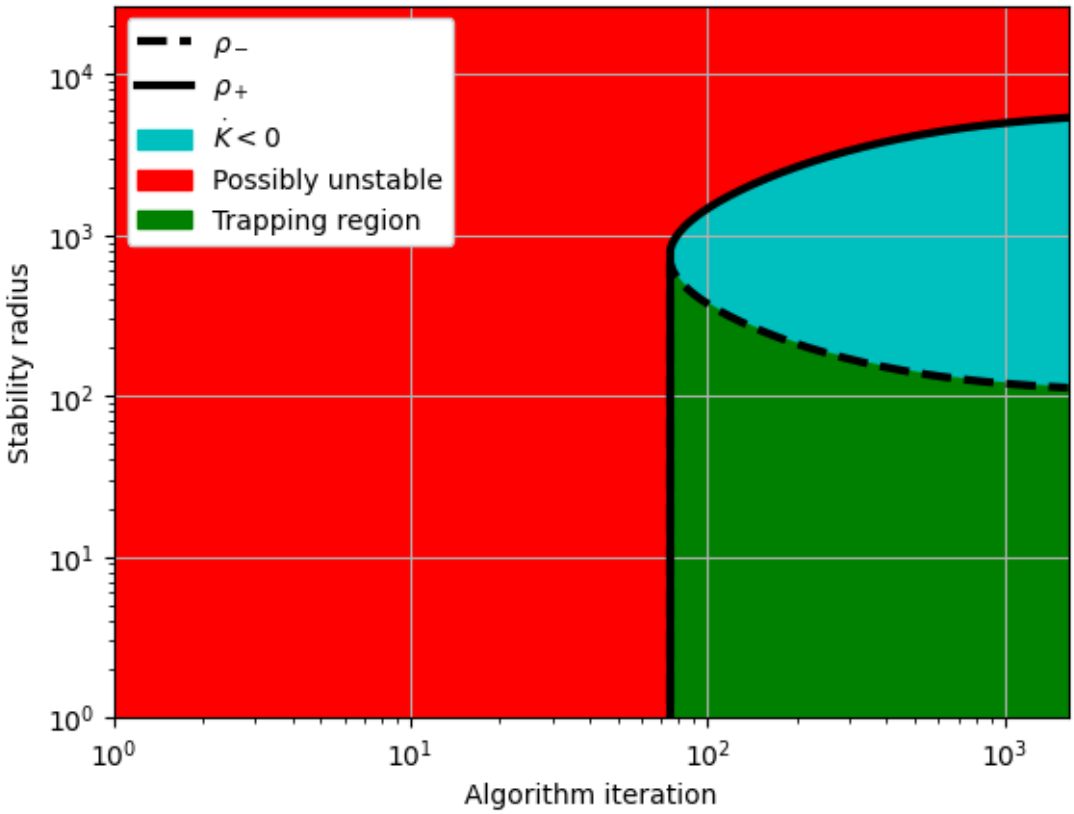

Figure 4.2: The progress of stability radius growth in training Lorenz system. The green area below the dashed line is identified as the smallest ball-shape trapping region found by the model and the blue area between the solid line and the dashed line corresponds to the dotted area in Fig. 3.3, where the derivative of the kinetic energy $\dot{K}$ is negative, so all trajectories starting inside this area will finally fall into the green one. The red area outside means no guarantees for the stability and possibly unstable.

We also demonstrate the results qualitatively with figures of trajectories of the model for some examples to examine, for instance, the noise robustness of the computational model. Another reason to exhibit results qualitatively rather than quantitatively is that quantifying the model performance for chaotic systems are very challenging because of its sensitive dependence on initial conditions and aperiodic long-term behavior. However, despite the different ways we present for each example, we conduct the experiment in quite similar approach. For each system, a single trajectory whose initial condition is arbitrarily chosen, is trained by SINDy with the extended trapping optimizer and keeping the same temporal duration, we then evaluate the model on a different trajectory with a new random initial condition. Table 4.2 summarizes the sampling, hyperparameters, and identified local stability estimates for a number of examples discussed in Sec. 4.2.1-4.2.2. Several chaotic systems with different strange attractors will be presented in this section. Note that it is usually difficult to quantify model performance for chaotic system, so we utilize Eq. (4.7) to investigate the utility of our extended trapping algorithm on these systems rather than calculating the time-average error in $\dot{\boldsymbol{X}}$.

The local stability size ρ_+ and the smallest trapping region size ρ_- for each model are calculated if applicable for evaluation purpose. It is usually expected that we should observe the increase of the ρ_+ and the decrease of ρ_- as the iteration goes as shown in Fig. 4.2, which well illustrates Thm. 3.2 and the two limits (3.26). To compare the smallest trapping region size across different systems we identify, a normalized quantity defined as $R_{eff} = \rho_- / \sqrt{\sum_{i=1}^{r} \bar{y}_i^2}$ is reported as well.

In this work $\boldsymbol{P} = \boldsymbol{I}$. So we have $\tilde{\boldsymbol{A}}^S = \boldsymbol{A}^S$ in all examples presented below.

4.2.1 Noisy Lorenz attractor

The well-known chaotic system developed by Lorenz in 1963 [32] has been interested researchers for decades. General findings about the existence of monotonically attracting trapping region are shown for quadratically nonlinear systems in Lorenz's work. Swinnerton-Dyer shows an attracting trapping region for the Lorenz system via the existence of a Lyapunov function outside the trapping region [54]. The globally stable chaotic system

Table 4.2: Description of the sampling, extended trapping SINDy hyperparameters and identified local stability estimates for the dynamic systems examined in Sec. 4.2. ϵ_Q reports the deviation from $\boldsymbol{Q}$ having zero totally symmetric part. λ_1 denotes the largest eigenvalues in each $\tilde{\boldsymbol{A}}^S$ of the system.

Dynamic system	r	Δt	β	η	ϵ_Q	ρ_-	ρ_+	R_{eff}	λ_1	E_m	E_f
Lorenz Attractor	3	0.01	1.78×10^{-7}	10^{-4}	1.3×10^{-3}	200	221	8.3	-0.95	1.5	10^{-5}
10% noisy Lorenz	3	0.01	2.27×10^{-9}	10^{-4}	1.1×10^{-3}	220	243	9.3	-0.92	1.1	0.06
50% noisy Lorenz	3	0.01	10^{-10}	10^{-4}	0.01				0.05	1.3	0.9
Finance	3	0.1	10^{-9}	10^{-3}	1.1×10^{-9}		5.4×10^{7}	32.7	-0.1		
Hadley	3	0.015	10^{-9}	10^{-3}	3.5×10^{-10}		1.7×10^{8}	22	-0.1		
LorenzStenflo	4	0.039	10^{-9}	10^{-3}	6.1×10^{-7}		6×10^{4}	184	-0.1		
VallisElNino	3	0.022	10^{-9}	10^{-3}	3.3×10^{-7}		1.7×10^{5}	3.8×10^{3}	-0.1		

reads,

$$\dot{x} = \sigma(y - x), \tag{4.9a}$$

$$\dot{y} = x(\rho - z) - y, \tag{4.9b}$$

$$\dot{z} = xy - \beta z. \tag{4.9c}$$

We choose the same parameters as in Lorenz 1963 [32], which are $\sigma = 10$, $\rho = 28$, $\beta = 8/3$. And we add 50% noise to the data with the choice of Lyapunov matrix $\boldsymbol{P}$ being the identity matrix $\boldsymbol{I}$ to examine the noise robustness of extended trapping algorithm. The noise is referred to as Gaussian noise, whose standard deviation is given by the root mean square of the test trajectory and the mean is zero. So the 10% noise denotes that the standard deviation is 10% of the root mean square as shown in Fig. 4.4. For comparison, we also show the identified Lorenz system without noise added as shown in Fig. 4.3.

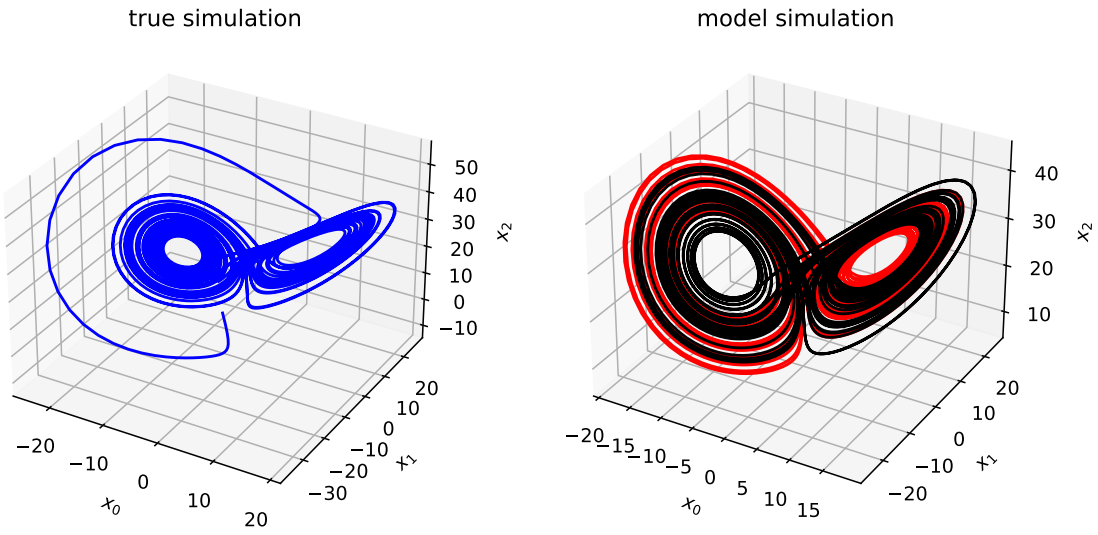

Figure 4.3: Lorenz system without noise.

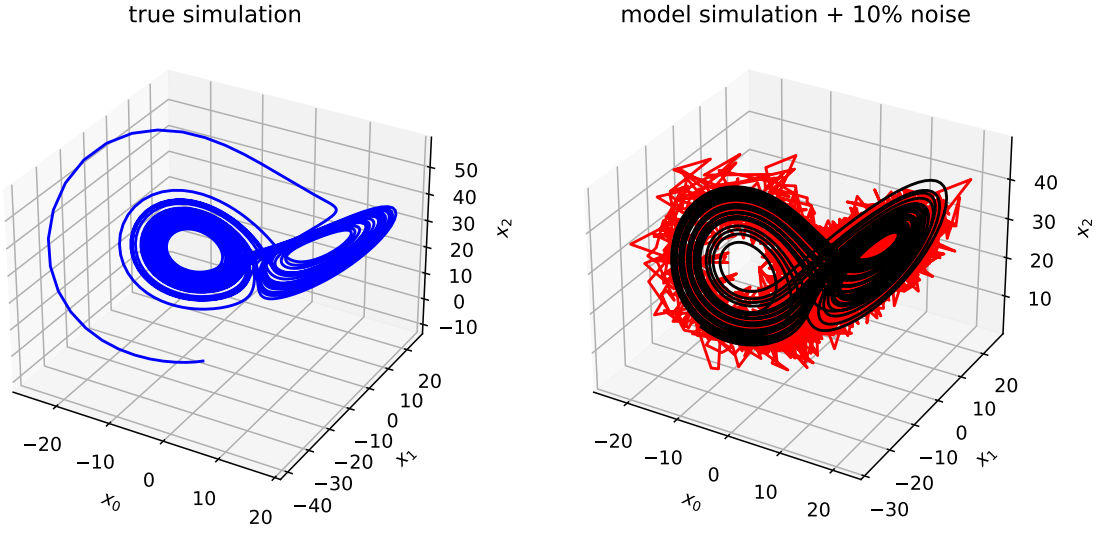

Figure 4.4: 10% noisy Lorenz.

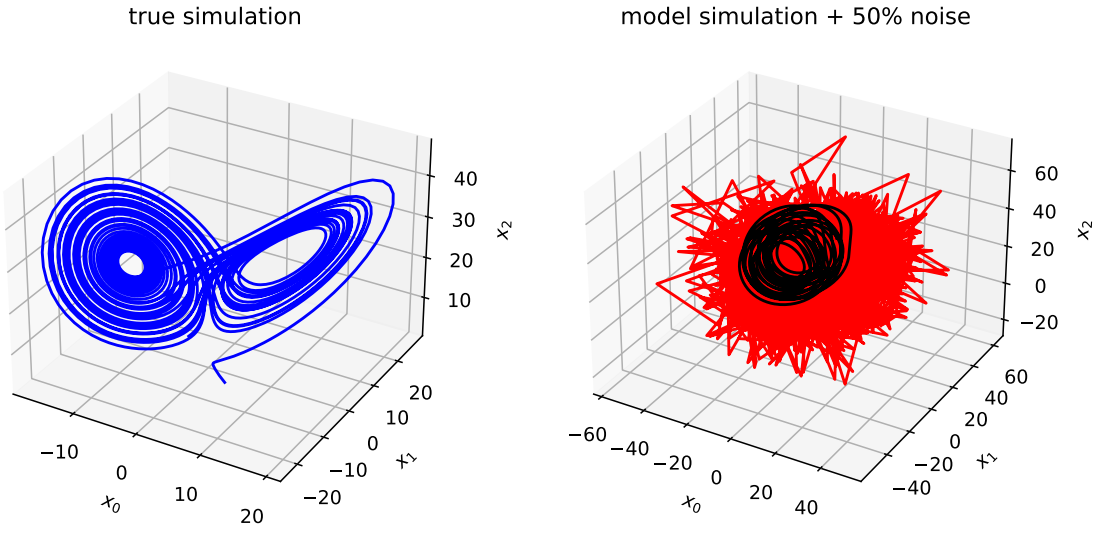

Figure 4.5: 50% noisy Lorenz.

Results have shown that the algorithm can improve robustness to noise. With 10% noise added, the attractor can be well identified, and with 50% noise added, the model identified is still stable and presents a different strange attractor, although fails to report a locally stable region. Another observation is that the $\boldsymbol{A}^S$ of this 50% noise model is not *Hurwitz*, which has one and only one positive eigenvalue. This implies that even if the $\boldsymbol{A}^S$ is not negative-definite, the quadratically nonlinear system may still be locally stable, but the condition for its being locally stable is not clear. Indeed, it may not be fully locally stable, but simply have only a very small or degenerate set of initial conditions that can in principle escape to infinity.

4.2.2 Assorted locally stable models from the dysts database

The dysts dataset [16] is a collection of more than 100 chaotic system, four of them that has energy-preserving structure in the quadratic terms are picked for benchmark testing. These datasets of all systems are normalized so we can compare the performance of one set of combinations of hyperparameters on different dynamical systems. And as it shows, the algorithm successfully recognizes the attractors and performs well on promoting stability

in modeling these systems. Table 4.2 also gives the normalized trapping region size of each system.

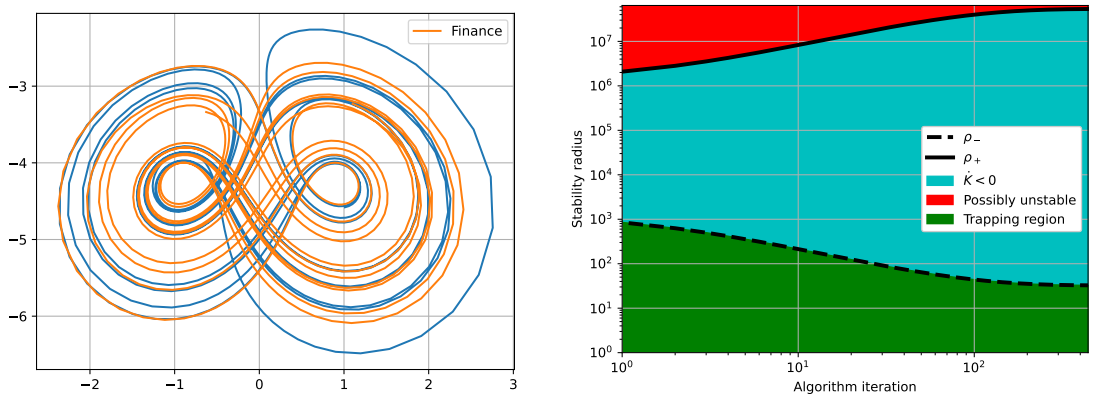

Figure 4.6: A finance chaotic system.

The finance chaotic system shown in Fig. 4.6 is found by Cai *et al.* [8], and described as

$$\dot{x} = z + (y - \alpha)x, \tag{4.10a}$$

$$\dot{y} = 1 - \beta y - x^2, \tag{4.10b}$$

$$\dot{z} = -x - \sigma z, \tag{4.10c}$$

where the parameters are $\alpha = 10^{-5}$, $\beta = 0.1$ and $\sigma = 1$.

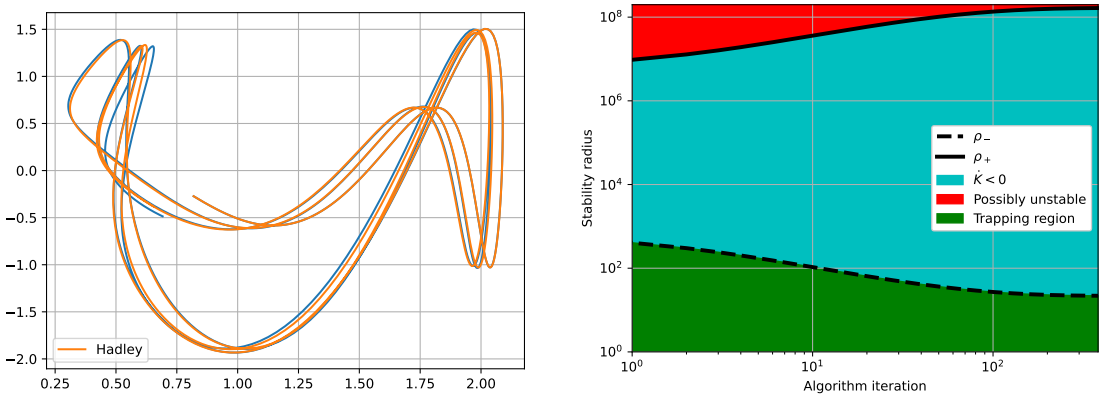

Figure 4.7: Hadley cell dynamics.

The Hadley system is an atmospheric convective cell which is also known as Hadley cell or Hadley circulation [19]. It was first envisioned in 1735 to explain the trade winds by Hadley. The result of a 3D model for this test is shown in Fig. 4.7.

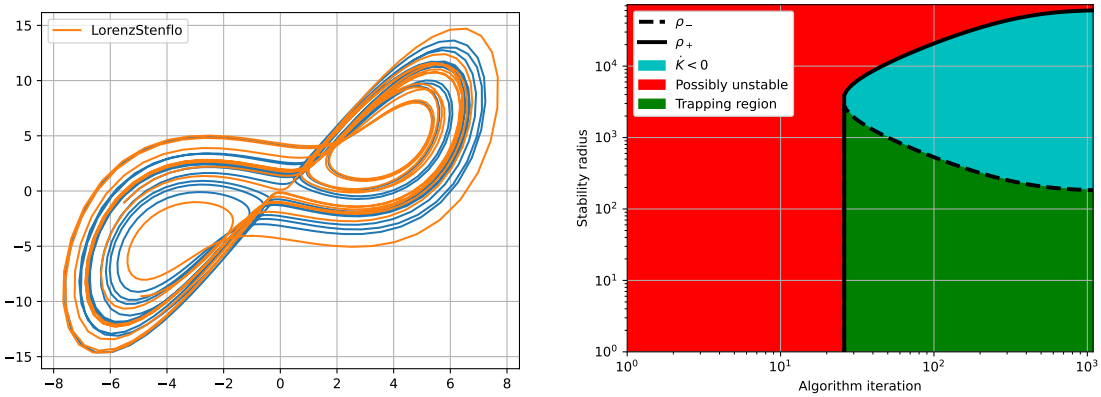

Figure 4.8: Stochastic Lorenz–Stenflo system.

Stochastic Lorenz–Stenflo system is a 4D system describing atmospheric acoustic-gravity waves by Stenflo [52]. The system reads,

$$\dot{x} = \sigma(y - x) + sw, \tag{4.11a}$$

$$\dot{y} = rx - xz - y, \tag{4.11b}$$

$$\dot{z} = xy - bz, \tag{4.11c}$$

$$\dot{w} = -x - \sigma w, \tag{4.11d}$$

where the parameters are $\sigma = 1$, $b = 0.7$, $r = 26$ and $s = 1.5$.

El Niño phenomenon has a great impact on the global climate and a simple model to describe it is Vallis continuous-time model [58]. The model is in a form of 3D quadratically nonlinear system of ODE and has three parameters:

$$\dot{x} = By - C(x + p), \tag{4.12a}$$

$$\dot{y} = -y + xz, \tag{4.12b}$$

$$\dot{z} = xy - bz, \tag{4.12c}$$

$$\dot{w} = -z - xy + 1, \tag{4.12d}$$

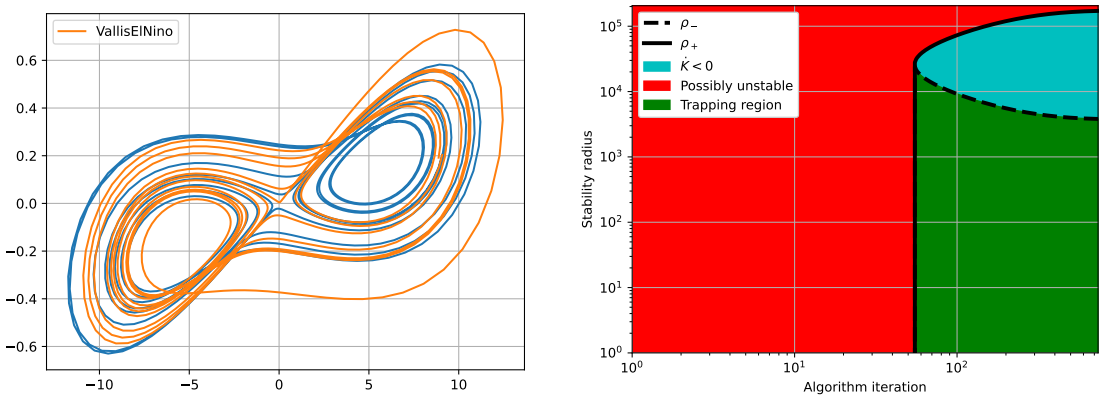

Figure 4.9: Vallis model for El Niño.

where $B = 102$, $C = 3$ and $p = 0$ results in a chaotic system shown in Fig. 4.9.

4.2.3 Lid-cavity flow

The fluid mechanical system of lid-driven cavity is commonly used as a benchmark for testing numerical methods [28]. The system is set as that the viscous fluid is in a confined geometry of a rectangular or a cubic container and its motion is driven by the tangentially moving of a bounding wall. We consider the incompressible flow of a Newtonian fluid in this problem, so that the fluid motion is governed by the incompressible Navier-Stokes equations. We then apply the model reduction method previously mentioned, resulting in a POD-Galerkin model. In this work, we select first 6 POD modes to build the data-driven model.

Then the extended trapping SINDy algorithm is applied to compare with direct numerical simulation, 6-modes POD-Galerkin model, 16-modes POD-Galerkin model and a naive SINDy model. As shown in Fig. 4.10, the extended trapping SINDy model outperforms all other presented data-driven models and indicates strong stability compared to all others, even though the $\boldsymbol{A}^S$ from the extended trapping SINDy is not fully promoted to negative-definite. The $\boldsymbol{A}^S$ has one positive eigenvalue which is similar as the one in the 50% noisy Lorenz example in Sec. 4.2.1.

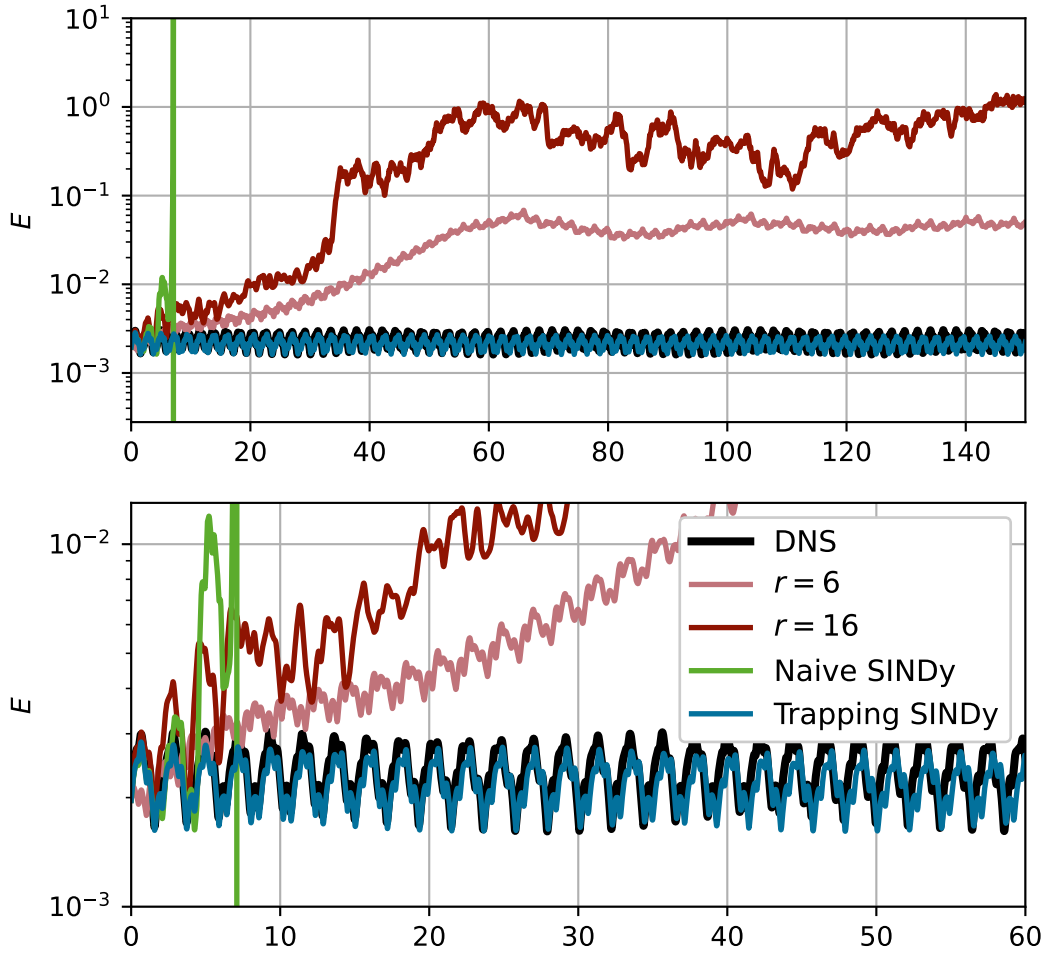

Figure 4.10: Lid-driven cavity. E denotes the total kinetic energy. Extended trapping SINDy and naive SINDy both use the first 6 POD modes.

Chapter 5
CONCLUSIONS AND FUTURE WORK

Motivated by the nonlinear energy-preserving structure in multiple dynamical systems of fluids and plasmas, as well as the weakly nonlinear energy-preserving constraints in open-channel flows, this thesis has considered the boundedness issue in quadratically nonlinear models and provide a new theorem with explicit bounds on the size of trapping regions for these models. An algorithm was proposed to identify such models from data based on this theorem.

In Chap. 3, we analyzed the stability of linear-quadratic systems with a weakly energy-preserving constraints on nonlinear terms. By adopting Lyapunov's direct method, a new theorem, which includes the sufficient conditions for the existence of trapping regions in systems with such constraints and the estimates of the size of these trapping regions, was proved. We also compared our theorem with Schlegel and Noack trapping theorem, which we showed that ours can exactly reproduce their trapping theorem.

In Chap. 4, based on the trapping SINDy algorithm by Kaptanoglu *et al.*, an extended trapping SINDy algorithm was proposed to promote local stability of data-driven quadratically nonlinear models. One of the guarantees on performance needed to use this algorithm is given by the local stability estimates from the theorem proposed in Chap. 3. The algorithm performs well on tasks such as system identification and simulation with ROMs.

Successfully promoting the local stability of data-driven models requires a combination of well designed algorithms, methods of analysis with physical insight and interpretations, and local or global stability guarantees. The work of this thesis was motivated, in part, by the idea that tools that extract models from data should guarantee the stability of models that capture the dominant behavior of bounded physical systems like fluids which can be described by Navier-Stokes equations. Implicit in this motivation is the fact that accurately identifying reduced-order models is not necessarily the final goal, which is being able to

provide less conservative and better mathematical description of the stability estimates by general and feasible analytical methods for systems with constraints and well-designed algorithms to achieve the data-driven reduced-order models. This leads to a number of possible directions for future work that would extend the research presented in this thesis:

- **Finding provably unbounded initial conditions:** Even though we have given sufficient conditions for attracting trapping regions to exist in quadratically nonlinear systems and conservative estimates about the size of the local stability domain, an unbounded trajectory has not been yet found when the *Hurwitz* property is slightly broken. It might be illustrative to find provably unbounded initial conditions starting arbitrarily close to the origin for Lorenz or similar systems that are analytically tractable when $\boldsymbol{A}^S$ is not *Hurwitz*.

- **Improve the strategy of enforcing a Hurwitz matrix $\boldsymbol{A}^S$ in trapping SINDy algorithm:** The strategy to enforce a *Hurwitz* matrix $\boldsymbol{A}^S$ is to solve $\boldsymbol{\xi}$ and $\boldsymbol{m}$ separately in each iteration so that each process is convex. However, this convex composite term indeed makes the loss function nonconvex. The convergence properties are unclear as well. A promising future direction is to improve the approaches to solve these nonconvex problem with such convex-composite term.

- **Data-driven methods to find the optimal Lyapunov matrix $\boldsymbol{P}$ to enlarge the guaranteed stability region:** From the Lyapunov equation (2.23) and Lyapunov function (2.20) we can observe that it is possible to control the magnitudes of the eigenvalues of $\tilde{\boldsymbol{A}}^S$ using different Lyapunov matrix $\boldsymbol{P}$. This implies that some $\boldsymbol{P}$ can enlarge the the guaranteed stability region by forming $\tilde{\boldsymbol{A}}^S$ with an even bigger largest eigenvalue so that ρ_+ is bigger in this way. Searching for an optimal $\boldsymbol{P}$ through data-driven methods especially machine learning algorithms is promising to extend the algorithm in this thesis even further.

- **Extending the theorem developed in this work to system identification with deep learning:** We have proved that the methods proposed in this thesis

can be used in system identification but the problem is nonconvex. With the help of modern powerful computational platforms, it can be solved by implementing this method in deep learning. So developing a deep neural network for this method is another promising future work. We can extend other data-driven machine learning techniques for fluids with the proposed theorem as well.

BIBLIOGRAPHY

[1] Peter Benner and Tobias Breiten. Two-sided projection methods for nonlinear model order reduction. SIAM Journal on Scientific Computing, 37(2):B239–B260, 2015.

[2] Peter Benner and Tobias Breiten. Two-sided projection methods for nonlinear model order reduction. SIAM Journal on Scientific Computing, 37(2):B239–B260, 2015.

[3] Peter Benner, Serkan Gugercin, and Karen Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. SIAM review, 57(4):483–531, 2015.

[4] Josh C. Bongard and Hod Lipson. Automated reverse engineering of nonlinear dynamical systems. Proceedings of the National Academy of Sciences, 104:9943 – 9948, 2007.

[5] Steven L Brunton. Applying machine learning to study fluid mechanics. Acta Mechanica Sinica, 37(12):1718–1726, 2021.

[6] Steven L Brunton and J Nathan Kutz. Data-driven science and engineering: Machine learning, dynamical systems, and control. Cambridge University Press, 2019.

[7] Steven L. Brunton, Joshua L. Proctor, and J. Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proceedings of the National Academy of Sciences, 113(15):3932–3937, 2016.

[8] Guoliang Cai and Juanjuan Huang. A new finance chaotic attractor. International Journal of Nonlinear Science, 3(3):213–220, 2007.

[9] Kevin Carlberg, Charbel Bou-Mosleh, and Charbel Farhat. 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):155–181, 2011.

[10] Kevin Carlberg, Charbel Farhat, Julien Cortial, and David Amsallem. The GNAT method for nonlinear model reduction: effective implementation and application to computational fluid dynamics and turbulent flows. Journal of Computational Physics, 242:623–647, 2013.

[11] Kathleen Champion, Bethany Lusch, J Nathan Kutz, and Steven L Brunton. Data-driven discovery of coordinates and governing equations. Proceedings of the National Academy of Sciences, 116(45):22445–22451, 2019.

[12] Kathleen Champion, Peng Zheng, Aleksandr Y. Aravkin, Steven L. Brunton, and J. Nathan Kutz. A unified sparse optimization framework to learn parsimonious physics-informed models from data. IEEE Access, 8:169259–169271, 2020.

[13] Brian M. de Silva, Kathleen Champion, Markus Quade, Jean-Christophe Loiseau, J. Nathan Kutz, and Steven L. Brunton. Pysindy: A python package for the sparse identification of nonlinear dynamical systems from data. Journal of Open Source Software, 5(49):2104, 2020.

[14] AE Deane, IG Kevrekidis, G Em Karniadakis, and SA Orszag. Low-dimensional models for complex geometry flows: application to grooved channels and circular cylinders. Physics of Fluids A: Fluid Dynamics, 3(10):2337–2354, 1991.

[15] Jeffrey P Freidberg. Ideal MHD. Cambridge University Press, 2014.

[16] William Gilpin. Chaos as an interpretable benchmark for forecasting and data-driven modelling, 2023.

[17] Sebastian Grimberg, Charbel Farhat, and Noah Youkilis. On the stability of projection-based model order reduction for convection-dominated laminar and turbulent flows. Journal of Computational Physics, 419:109681, 2020.

[18] Chenjie Gu. 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):1307–1320, 2011.

[19] George Hadley. Vi. concerning the cause of the general trade-winds. Philosophical Transactions of the Royal Society of London, 39(437):58–62, 1735.

[20] Philip Holmes, John L Lumley, Gahl Berkooz, and Clarence W Rowley. Turbulence, coherent structures, dynamical systems and symmetry. Cambridge university press, 2012.

[21] Alan A Kaptanoglu, Tom E Benedett, Kyle D Morgan, Chris J Hansen, and Thomas R Jarboe. Two-temperature effects in Hall-MHD simulations of the HIT-SI experiment. Physics of Plasmas, 27(7):072505, 2020.

[22] Alan A Kaptanoglu, Jared L Callaham, Aleksandr Aravkin, Christopher J Hansen, and Steven L Brunton. Promoting global stability in data-driven models of quadratic nonlinear dynamics. Physical Review Fluids, 6(9):094401, 2021.

[23] Alan A. Kaptanoglu, Brian M. de Silva, Urban Fasel, Kadierdan Kaheman, Andy J. Goldschmidt, Jared Callaham, Charles B. Delahunt, Zachary G. Nicolaou, Kathleen Champion, Jean-Christophe Loiseau, J. Nathan Kutz, and Steven L. Brunton. PySINDy: A comprehensive Python package for robust sparse system identification. Journal of Open Source Software, 7(69):3994, 2022.

[24] Alan A Kaptanoglu, Kyle D Morgan, Chris J Hansen, and Steven L Brunton. Physics-constrained, low-dimensional models for magnetohydrodynamics: First-principles and data-driven approaches. Physical Review E, 104(1):015206, 2021.

[25] H.K. Khalil. Nonlinear Systems. Pearson Education. Prentice Hall, 2002.

[26] Boris Kramer. Stability domains for quadratic-bilinear reduced-order models. SIAM Journal on Applied Dynamical Systems, 20(2):981–996, 2021.

[27] Boris Kramer and Karen E Willcox. Nonlinear model order reduction via lifting transformations and proper orthogonal decomposition. AIAA Journal, 57(6):2297–2307, 2019.

[28] Hendrik C Kuhlmann and Francesco Romanò. The lid-driven cavity. Computational Modelling of Bifurcations and Instabilities in Fluid Dynamics, pages 233–309, 2019.

[29] Jean-Christophe Loiseau and Steven L. Brunton. Constrained sparse Galerkin regression. Journal of Fluid Mechanics, 838:42–67, March 2018.

[30] Jean-Christophe Loiseau, Steven L Brunton, and Bernd R Noack. From the POD-Galerkin method to sparse manifold models. Handbook of Model-Order Reduction, 2:1–47, 2019.

[31] Jean-Christophe Loiseau, Bernd R Noack, and Steven L Brunton. Sparse reduced-order modeling: sensor-based dynamics to full-state estimation. Journal of Fluid Mechanics, 844:459–490, 2018.

[32] Edward N Lorenz. Deterministic nonperiodic flow. Journal of atmospheric sciences, 20(2):130–141, 1963.

[33] David J. Lucia, Philip S. Beran, and Walter A. Silva. Reduced-order modeling: new approaches for computational physics. Progress in Aerospace Sciences, 40(1):51–117, 2004.

[34] William D McComb. The physics of fluid turbulence. Oxford, 1990.

[35] J.D. Meiss. Differential Dynamical Systems. Society for Industrial and Applied Mathematics, 2007.

[36] Francisco J. Montáns, Francisco Chinesta, Rafael Gómez-Bombarelli, and J. Nathan Kutz. Data-driven modeling and learning in science and engineering. Comptes Rendus Mécanique, 347(11):845–855, 2019. Data-Based Engineering Science and Technology.

[37] Bernd R Noack, Konstantin Afanasiev, Marek Morzyński, Gilead Tadmor, and Frank Thiele. A hierarchy of low-dimensional models for the transient and post-transient cylinder wake. Journal of Fluid Mechanics, 497:335–363, 2003.

[38] Bernd R Noack, Marek Morzynski, and Gilead Tadmor. Reduced-order modelling for flow control, volume 528. Springer Science & Business Media, 2011.

[39] Said Ouala, Steven L. Brunton, Bertrand Chapron, Ananda Pascual, Fabrice Collard, Lucile Gaultier, and Ronan Fablet. Bounded nonlinear forecasts of partially observed geophysical systems with physics-constrained deep learning. Physica D: Nonlinear Phenomena, 446:133630, 2023.

[40] Elizabeth Qian, Boris Kramer, Benjamin Peherstorfer, and Karen Willcox. Lift & learn: Physics-informed machine learning for large-scale nonlinear dynamical systems. Physica D: Nonlinear Phenomena, 406:132401, 2020.

[41] Cedric Raibaudo, Peng Zhong, Bernd R Noack, and Robert John Martinuzzi. Machine learning strategies applied to the control of a fluidic pinball. Physics of Fluids, 32(1), 2020.

[42] Dietmar Rempfer and Hermann F Fasel. Dynamics of three-dimensional coherent structures in a flat-plate boundary layer. Journal of Fluid Mechanics, 275:257–283, 1994.

[43] Feng Ren, Hai bao Hu, and Hui Tang. Active flow control using machine learning: A brief review. Journal of Hydrodynamics, 32:247–253, 2020.

[44] Clarence W Rowley and Scott TM Dawson. Model reduction for flow analysis and control. Annual Review of Fluid Mechanics, 49:387–417, 2017.

[45] Bernd Rummler and Andreas Noske. Direct Galerkin approximation of plane-parallel-Couette and channel flows by Stokes eigenfunctions. Notes on numerical fluid mechanics, 64:3–19, 1998.

[46] Michael A Savageau and Eberhard O Voit. Recasting nonlinear differential equations as s-systems: a canonical nonlinear form. Mathematical biosciences, 87(1):83–115, 1987.

[47] Nihar Sawant, Boris Kramer, and Benjamin Peherstorfer. Physics-informed regularization and structure preservation for learning stable reduced models from data with operator inference. Computer Methods in Applied Mechanics and Engineering, 404:115836, 2023.

[48] Michael Schlegel and Bernd R. Noack. On long-term boundedness of galerkin models. Journal of Fluid Mechanics, 765:325–352, Feb 25 2015.

[49] Hermann Schlichting and Klaus Gersten. Boundary-layer theory. Springer, 2016.

[50] Michael Schmidt and Hod Lipson. Distilling free-form natural laws from experimental data. Science, 324(5923):81–85, 2009.

[51] D. Sipp and A. Lebedev. Global stability of base and mean flows: a general approach and its applications to cylinder and open cavity flows. Journal of Fluid Mechanics, 593:333–358, 2007.

[52] L Stenflo. Generalized lorenz equations for acoustic-gravity waves in the atmosphere. Physica Scripta, 53(1):83, 1996.

[53] Steven H Strogatz. Nonlinear dynamics and chaos with student solutions manual: With applications to physics, biology, chemistry, and engineering. CRC press, 2018.

[54] Peter Swinnerton-Dyer. Bounds for trajectories of the lorenz equations: an illustration of how to choose liapunov functions. Physics Letters A, 281(2):161–167, 2001.

[55] Kunihiko Taira, Steven L Brunton, Scott Dawson, Clarence W Rowley, Tim Colonius, Beverley J McKeon, Oliver T Schmidt, Stanislav Gordeyev, Vassilios Theofilis, and Lawrence S Ukeiley. Modal analysis of fluid flows: An overview. AIAA Journal, 55(12):4013–4041, 2017.

[56] Kunihiko Taira, Maziar S Hemati, Steven L Brunton, Yiyang Sun, Karthik Duraisamy, Shervin Bagheri, Scott TM Dawson, and Chi-An Yeh. Modal analysis of fluid flows: Applications and outlook. AIAA journal, 58(3):998–1022, 2020.

[57] Robert Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1):267–288, 1996.

[58] GK Vallis. Conceptual models of el niño and the southern oscillation. Journal of Geophysical Research: Oceans, 93(C11):13979–13991, 1988.

[59] Sagar Verma. A survey on machine learning applied to dynamic physical systems. ArXiv, abs/2009.09719, 2020.

[60] Zhong Yi Wan and Themistoklis P Sapsis. Machine learning the kinematics of spherical particles in fluid flows. Journal of Fluid Mechanics, 857:R2, 2018.

[61] Peng Zheng and Aleksandr Aravkin. Relax-and-split method for nonconvex inverse problems. Inverse Problems, 36(9):095013, sep 2020.

[62] Yu Zhou, Dewei Fan, Bingfu Zhang, Ruiying Li, and B. R. Noack. Artificial intelligence control of a turbulent jet. Journal of Fluid Mechanics, 897, 2018.