Data-Driven Methods for System Identification and Lyapunov Stability

by

Thanin Quartz

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Applied Mathematics

Waterloo, Ontario, Canada, 2023

© Thanin Quartz 2023

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

I am the author of Chapter 2 and the theoretical contributions of Chapter 3 that is based on the NeurIPS paper [46]. My colleague Ruikun Zhou wrote the code for the numerical experiments contained in that section. The first part of the material in Chapter 4 is a review of the work already established in the community but authored by myself. The material in Chapters 4.3.3 and 4.4 is my novel contribution to the best of my knowledge. The material in Chapter 5 is a review of recent results in the computational dynamical systems field and is included as it meshes well with Chapter 3.

Abstract

This thesis focuses on data-driven methods applied to system identification and stability analysis of dynamical systems. In the first major contribution of the theorem we propose a learning framework to simultaneously stabilize an unknown nonlinear system with a neural controller and learn a neural Lyapunov function to certify a region of attraction (ROA) for the closed-loop system. The algorithmic structure consists of two neural networks and a satisfiability modulo theories (SMT) solver. The first neural network is responsible for learning the unknown dynamics. The second neural network aims to identify a valid Lyapunov function and a provably stabilizing nonlinear controller. The SMT solver then verifies that the candidate Lyapunov function indeed satisfies the Lyapunov conditions. We provide theoretical guarantees of the proposed learning framework in terms of the closed-loop stability for the unknown nonlinear system. We illustrate the effectiveness of the approach with a set of numerical experiments. We then examine another popular data driven method for system identification involving the Koopman operator. Methods based on the Koopman operator aim to approximate advancements of the state under the flow operator by a high-dimensional linear operator. This is accomplished by the extended mode decomposition (eDMD) algorithm which takes non-linear measurements of the state. Under the suitable conditions we have a result on the weak convergence of the eigenvalues and eigenfunctions of the eDMD operator that can serve as components of Lyapunov functions. Finally, we review methods for finding the region of attraction of an asymptotically stable fixed point and compare this method to the two methods mentioned above.

Acknowledgements

First of all, I would like to express my sincere gratitude and deep appreciation to my supervisors, Dr. Hans De Sterck and Dr. Jun Liu, for their guidance and timely advice. This work could not have been accomplished without their support, dedication and inspiration.

Secondly, I would like to acknowledge Dr. Giang Tran and Dr. Xinzhi Liu for their time spent on my thesis examining committee and valuable revisions.

I would like to thank all members of the hybrid system lab, especially my co-author Ruikun. I would also like to thank Greg and Somayeh for being regular members of the scientific computing lab. Last but not least, I would like to thank all my friends across the various departments and labs that enriched my Master's experience, especially to those that purchased Pokemon cards for my birthday. The money you wasted on cardboard provided me with immense happiness. Thank you all for giving me valuable suggestions for my research and holding me accountable. In particular, your company and warmth prevented me from freezing during my first Canadian winter.

Table of Contents

List of Figures viii

List of Tables ix

1 Introduction 1

2 Background and Notation 3

2.1 Dynamical Systems 3

2.2 Control Theory 4

2.3 Neural Network Theory 5

3 Identification of Lyapunov Functions with Neural Networks 7

3.1 Existence of neural Lyapunov functions 8

3.2 Learn and Stabilize Dynamics with Neural Lyapunov Functions 11

3.2.1 Learning Unknown Dynamics 12

3.2.2 Neural Lyapunov Function and Nonlinear Controller 12

3.2.3 Asymptotic Stability Guarantees of Unknown Nonlinear Systems . . 16

3.3 Experiments 18

3.3.1 Van der Pol Oscillator 18

3.3.2 Unicycle Path Following 21

3.3.3 Inverted Pendulum. 22

4 Koopman Operator **24**

4.1 Eigenfunctions and Spectrum of the Eigenvalues 25

4.2 Dynamic Mode Decomposition 27

4.3 Extended DMD and Convergence 30

4.3.1 Convergence of the eDMD Operator 33

4.3.2 Simultaneous Convergence of the eDMD Operator 37

4.3.3 Application of the Convergence of the eDMD Operator to Stability Analysis 41

4.4 Infinite Matrix Representation of the Koopman Operator 42

5 Estimating the Region of Attraction **45**

5.1 Theoretical Foundations 46

5.2 Methodology 49

5.3 Numerical Implementation 50

5.3.1 Van Der Pol Oscillator 51

5.3.2 Chiang and Thorp Differential Equation 53

6 Conclusion and Future Work **56**

6.1 Conclusion 56

6.2 Future Work 57

References **59**

APPENDICES **64**

A Measure Theory **65**

A.1 Primer on Measure spaces 65

A.2 Projections on Measure Spaces 66

B Proof of Theorem 6 **68**

List of Figures

3.1 The schematic diagram of the proposed algorithm. 11

3.2 Algorithmic structure of learning a neural Lyapunov function and the corresponding nonlinear controller with a 1-hidden layer neural network and an SMT solver. 13

3.3 Phase space plot and the limit cycle (bold black line) of Val Der Pol oscillator without controller, where the area within the bold black curve forms the actual ROA. 19

3.4 Neural Lyapunov function and the corresponding estimated ROA for Van der Pol oscillator. 20

3.5 Comparison of obtained ROAs for path following and inverted pendulum. . 22

5.1 Phase space plot of the Val Der Pol oscillator without controller. From this image we can form an idea for the actual ROA. 51

5.2 Estimates of the ROA and boundary of different evolutionary time 52

5.3 The level set function $\phi(x, 10)$ and the phase portrait of the system. 53

5.4 (a) Estimates of the ROA and boundary of different evolutionary time. (b) The level set of the curve $\phi(x, 10)$ and the phase portrait of the system. . 55

List of Tables

3.1 Parameters in Van der Pol Oscillator case 20

3.2 Parameters in unicycle path following case 22

3.3 Parameters in inverted pendulum case 23

Chapter 1

Introduction

This thesis explores the application of data-driven methods to challenging problems in dynamical and control systems ranging from system identification to developing Lyapunov functions. This contribution is significant as there is no general approach for finding a Lyapunov function for non-linear systems with provable guarantees. Recent development of learning-based methods have shown their effectiveness in identifying unknown (or partly known) systems [9, 29]. However, simultaneously finding Lyapunov functions and nonlinear controllers for systems with unknown dynamics is still an open and active problem in control and robotics applications [15]. We provide an answer to this problem using neural networks and discuss finding Lyapunov functions using Koopman eigenfunctions.

In practice, the prevailing way of stabilizing non-linear dynamical systems is to linearize the system around the equilibrium point and formulate linear-quadratic regulators (LQR) problems to minimize any possible deviations from the deviation point. This method does provide a linear feedback control policy, however this method typically produces a stability guarantee in some small neighbourhood of the equilibrium point. This will inevitably produce conservative systems and explains why applications to agile robotic locomotion is difficult [28]. Outside of the linearization region, Lyapunov functions have to be constructed to validate stability. Many existing methods for computing Lyapunov functions rely on making polynomial approximation of the dynamics and subsequently searching sum-of-square polynomials through semidefinite programming (SDP) to serve as Lyapunov functions [27]. This methodology is well studied, however imposing polynomial dynamics poses a strict restriction on the system. Additionally, numerical sensitivity in the SDP method means that the Lyapunov conditions are difficult to satisfy. Therefore, in this thesis, we use the fact that neural networks have much greater expressive ability than polynomials to provide guarantee of the Lyapunov conditions while simultaneously learning the controller. In addition, in the absence of a non-linear controller we show that Koopman operator methods can successfully perform system identification and provide Lyapunov functions for the learned system.

This thesis is organized as follows. Background material on dynamical systems, neural networks and control systems is provided in Chapter 2. Chapter 3 details the methodology and uses recently advances in computational verification for simultaneously learning a non-linear controller and verifying a valid Lyapunov function in the valid region. To support the neural methodology, we state and prove existence theorems for neural Lyapunov functions and their properties. This chapter concludes with several examples highlighting the practicality of these results. An overview of Koopman operator theory is then detailed in the beginning of Chapter 4. We then introduce several data-driven techniques that perform system identification based on the Koopman operator and show how the Koopman eigenfunctions can serve as Lyapunov functions. The final part of the chapter connects the Koopman learning algorithms with neural networks to search for basis functions that give optimal performance. Chapter 5 reviews recent advancements in estimating the ROA through its connection with the sub-level set of the viscosity solution of a Hamilton-Jacobi equation. Finally, we conclude this thesis in Chapter 6.2 along with possible extensions of the topics covered in this thesis that may be of interest.

Chapter 2

Background and Notation

2.1 Dynamical Systems

Throughout this work, we consider an autonomous system of the form

$$\dot{x} = f(x), \quad x(0) = x_0, \tag{2.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$. Since we need to analyze the stability properties of the system for (2.1) in the presence of uncertainty (due to the need to approximate the unknown dynamics), we introduce a more general notion of stability about a closed set A. When $A = \{0\}$, this coincides with the standard notion of stability about an equilibrium point. Intuitively, set stability w.r.t. to A is measured by closeness and convergence of solutions to the set A. To this end, define the distance from x to A by $\|x\|_A = \inf_{y \in A} \|x - y\|$. We need the following notions from stability analysis to formulate the theories in this thesis.

Definition 1 (Forward Invariance)**.** *A set $\Omega \subset \mathbb{R}^n$ is said to be forward invariant for (2.1) if for $x_0 \in \Omega$ implies that $x(t) \in \Omega$ for all $t \geq 0$.*

Definition 2 (Region of Attraction)**.** *For a closed forward invariant set A that is UAS, a region of attraction is the set of initial conditions in $\mathcal{D}$ such that the solution for the closed-loop system (2.1) is defined for all $t \geq 0$ and $\|x(t)\|_A \to 0$ as $t \to \infty$.*

Remark 1. *In particular, The ROA is the largest set contained in $\mathcal{D}$ satisfying Definition 2.*

We need to define notions of stability that are necessary to the proofs in Chapter 3.

Definition 3 (Set stability)**.** *A closed set $A \subseteq \mathbb{R}^n$ is said to be uniformly asymptotically stable (UAS) for the closed-loop system (2.1), if the following two conditions are met:*
(1) (Uniform stability) For every $\epsilon > 0$, there exists a $\delta_\epsilon > 0$ such that $\|x(0)\|_A < \delta_\epsilon$ implies that $x(t)$ is defined for $t \geq 0$ and $\|x\|_A < \epsilon$ for any solution x of (2.1) for all $t \geq 0$; and
(2) (Uniform attractivity) There exists some $\rho > 0$ such that, for every $\epsilon > 0$, there exists some $T > 0$ such that $x(t)$ is defined for $t \geq 0$ and $\|x(t)\|_A < \epsilon$ for any solution $x(t)$ of (2.1) whenever $\|x(0)\|_A < \rho$ and $t \geq T$.

Definition 4 (Reachable Set)**.** *Let $R^t(x_0)$ denote the point $x(t)$ reached by the solution of (2.1) at time t starting at x_0. For $T \geq 0$ define the finite time horizon reachable set as*

$$R^{0\leq t\leq T}(x_0) = \cup_{0\leq t\leq T} R^t(x_0).$$

Similarly, for a set $W \subset \mathcal{D}$, define

$$R^{0\leq t\leq T}(W) := \cup_{x_0\in W} R^{0\leq t\leq T}(x_0).$$

Similarly, if solutions are defined for all $t \geq 0$, then the reachable set is defined as

$$R(W) := \cup_{x_0\in W} \cup_{t\geq 0} R^t(x_0).$$

The following result states that the finite time horizon reachable set is compact and can be found in [11].

Lemma 1. *Suppose that $K \subset \mathbb{R}^n$ is a compact set. Then the set $R^{0\leq t\leq T}(K)$ is compact for any $T \geq 0$.*

2.2 Control Theory

Control theory is useful for dealing with any system that exhibits feedback. The control of non-linear systems is a challenging task that is of paramount importance for applications such as flow control [4] and eradicating infectious diseases [31]. We can think of a dynamical system in the presence of control as

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

where $x \in \mathcal{D}$ is the state of the system, $\mathcal{D} \subseteq \mathbb{R}^n$ is an open set containing the origin, $u \in \mathcal{U} \subseteq \mathbb{R}^m$ and $f : \mathcal{D} \times \mathcal{U} \to \mathcal{D}$. For this thesis, we want to have a concept of lie derivatives in the presence of a controller.

Definition 5 (Lie Derivatives)**.** *The Lie derivative of a continuously differentiable scalar function $V : \mathcal{D} \to \mathbb{R}$ over a vector field f and a nonlinear controller u is defined as*

$$\nabla_f V(x) = \sum_{i=1}^{n} \frac{\partial V}{\partial x_i} \dot{x}_i = \sum_{i=1}^{n} \frac{\partial V}{\partial x_i} f_i(x, u). \tag{2.2}$$

The lie derivative measures the rate of change along the system dynamics.

The following is a standard Lyapunov theorem for the UAS property of a compact invariant set.

Theorem 1 (Sufficient Condition for UAS property)**.** *Consider the nonlinear system* (2.1)*. Let $A \subset \mathcal{D}$ be a compact invariant set of this system. Suppose there exists a continuously differentiable function $V : \mathcal{D} \to \mathbb{R}$ that is positive definite with respect to A, i.e.,*

$$V(x) = 0 \ \forall x \in A \ \textit{and} \ V(x) > 0 \ \forall x \in \mathcal{D} \setminus A, \tag{2.3}$$

and the lie derivative is negative definite with respect to A, i.e.

$$\nabla_f V(x) < 0 \ \forall x \in \mathcal{D} \setminus A. \tag{2.4}$$

Then, A is UAS for the system.

See [20, 40] for sufficiency and necessity of Lyapunov conditions for set stability under more general settings. The function V satisfying (2.3) and (2.4) is called a Lyapunov function with respect to A.

2.3 Neural Network Theory

In recent years neural networks have been employed in a wide variety of scientific domains and industrial applications including learning differential equations, image classification and recommendation systems. In practice, neural networks are entirely data-driven and require a well chosen loss function to guide the learning process to an optimal model. A neural network is dependent on its hyperparemeters, parameters that are chosen before training, which includes the number of layers, hidden nodes, activation functions, and the architecture that connects the nodes. In this thesis we consider the most standard architecture, the multilayer feedforward perceptron (MLP) model, which takes the form

$$F_\theta(\boldsymbol{x}) = \boldsymbol{W}_L \left(\sigma \left(\cdots \sigma \left(\boldsymbol{W}_1 \boldsymbol{x} + \boldsymbol{b}_1 \right) \cdots \right) \right) + \boldsymbol{b}_L, \tag{2.5}$$

where L denotes the number of layers, σ is the non-affine activation function, W_i, $i = 1, \dots, L$ is the so-called weight matrix and b_i, $i = 1, \dots, L$ is called the bias vector. Note that $\{\mathbf{W}_i, \, \mathbf{b}_i\}_{i=1}^{L}$ are optimized over during the learning process to ensure the smallest training error. The key theorem that supports the practicality of neural networks is the universal approximation theorem which asserts that any continuous function can be approximated uniformly on compacta. The proof of the following result can be found in [30].

Theorem 2. *Suppose $K \subset \mathbb{R}^n$ is a compact set and $f : K \to \mathbb{R}^m \in C(\mathbb{R}^n)$. Let $\sigma \in C(\mathbb{R})$ and suppose that σ is not a polynomial function. Then for every $\epsilon > 0$ there exists a single layer neural network of the form $F_\theta(\mathbf{x}) = \boldsymbol{W}_2 \left(\sigma\left(\boldsymbol{W}_1 \mathbf{x} + \mathbf{b}_1\right)\right)$ such that*

$$\sup_{x \in K} |F_\theta(\mathbf{x}) - f(\mathbf{x})|_\infty < \epsilon. \tag{2.6}$$

First we state an extension of the universal approximation theorem that states it is possible to simultaneously pointwisely approximate a function and its partial derivatives by a neural network. The proof of this result can be found in [30] and the piece of detailed analysis omitted in the end of this proof can be found in Section 3 of [19].

Theorem 3. *Let $K \subset \mathbb{R}^m$ be a compact set and suppose $f : K \to \mathbb{R}^n \in C^1(\mathbb{R}^n)$. Then, for every $\epsilon > 0$ there exists a neural network $\phi : K \to \mathbb{R}$ of the form $\phi(x) = \mathbf{W}_2(\sigma \circ (\mathbf{W}_1 x + \mathbf{b}))$ for $\sigma \in C^1(\mathbb{R})$ and not a polynomial, $\mathbf{W}_1 \in \mathbb{R}^{k \times m}, \mathbf{b} \in \mathbb{R}^k$ and $\mathbf{W}_2 \in \mathbb{R}^{k \times n}$ for some $k \in \mathbb{N}$ such that*

$$\|f - \phi\|_\infty := \sup_{x \in K} |f(x) - \phi(x)| < \epsilon \tag{2.7}$$

and for all $i = 1, \dots, n$, the following simultaneously holds

$$\left\| \frac{\partial f}{\partial x_i} - \frac{\partial \phi}{\partial x_i} \right\|_\infty < \epsilon. \tag{2.8}$$

Remark 2. *Note that the results stated above hold for deeper neural networks, but we state these results for neural networks with one hidden layer as this simplifies the exposition of the proofs in Chapter 3.*

Chapter 3

Identification of Lyapunov Functions with Neural Networks

The contents of this paper are based on my co-authored NeurIPS paper [46]. Throughout this chapter, we consider a nonlinear control system of the form

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

where $x \in \mathcal{D}$ is the state of the system, and $\mathcal{D} \subseteq \mathbb{R}^n$ is an open set containing the origin; $u \in \mathcal{K} \subseteq \mathbb{R}^m$ is the feedback control input given by $u = \kappa(x)$, where $\kappa(x)$ is a continuous function of x. Without loss of generality, we assume the origin is an equilibrium point of the closed-loop system

$$\dot{x} = f(x, \kappa(x)), \quad x(0) = x_0. \tag{3.2}$$

When there is no ambiguity, we also refer to the right-hand side of (3.2) by f.

We assume that we do not have explicit knowledge of the right-hand side of the system (3.1). The main objective is to stabilize the unknown dynamical system by designing a feedback control function κ. Stability guarantees are established using Lyapunov functions. We next present some preliminaries on model assumptions.

Assumption 1 (Lipschitz Continuity)**.** *The right-hand side of the nonlinear system (3.1) is assumed to be Lipschitz continuous, i.e.,*

$$\|f(x, u) - f(y, v)\| \leq L\|(x, u) - (y, v)\| \;\; \forall x, y \in \mathcal{D} \quad \textit{and} \quad \forall u, v \in \mathcal{U},$$

where L is called the Lipschitz constant; (x, u) and (y, v) denote the concatenation of the corresponding two vectors. We assume the Lipschitz constant L is known.

Remark 3. *In particular this ensures that the solution to (3.2) is unique and thus, the neural network responsible for system identifications will learn a unique function.*

Assumption 2 (Partly Known Dynamics)**.** *The linearized model about the origin in the form of $\dot{x} = Ax + Bu$, where A and B are constant matrices, is known for the nonlinear system (3.1).*

3.1 Existence of neural Lyapunov functions

A central problem in dynamics and control theory is the existence of Lyapunov functions. As a theoretical guarantee we show that it is possible to train a neural network as a practical Lyapunov function (i.e., set stability w.r.t. a sufficiently small neighborhood of the origin), provided that a Lyapunov function exists. According to converse Lyapunov theorems [20, 40], Lyapunov functions do exist when the origin is UAS. More specifically, we show that the learned neural network satisfies the Lyapunov conditions outside some neighborhood of the origin that can be chosen to be arbitrarily small in measure. The idea will be to perform an under-approximation of the domain $\mathcal{D}$ in a controlled way. Let $(\mathbb{R}^n, \mathcal{B}(\mathbb{R}^n), \mu)$ denote the standard measure space where $\mathcal{B}(\mathbb{R}^n)$ is the Borel σ-algebra and μ is the Lebesgue measure. The following lemma from [34] states that it is possible to arbitrarily under-approximate open sets by compact sets in measure.

Lemma 2. *For every open set $O \in \mathcal{B}(\mathbb{R}^n)$ such that $\mu(O) < \infty$ and every $\epsilon > 0$, there exists a compact set K such that $\mu(O \setminus K) < \epsilon$.*

By the pointwise approximation of the universal approximation theorem, it is not possible to satisfy the Lyapunov conditions on $\mathcal{D}$ as this set contains the origin. However, if there is a Lyapunov function for (3.2) that a neural network could learn and if practical stability is sufficient, Theorem 4 states that there exists a neural network satisfying the Lyapunov conditions on a compact set K except on a closed neighborhood B of the origin that is UAS. Moreover, this approximation can be controlled in measure.

Theorem 4. *Suppose that the origin is UAS for system (3.2) and $\mathcal{I}$ is a forward invariant set contained in the ROA of the origin. Fix any $\gamma_1, \gamma_2 > 0$. There exists a forward invariant and compact set $K \subset \mathcal{I}$ satisfying the under approximation $\mu(\mathcal{I} \setminus K) < \gamma_1$. On K there exists a neural network V_ϕ that satisfies the Lyapunov conditions on $K \setminus \mathcal{A}$, where $\mathcal{A}$ is a closed neighborhood of the origin. The neural Lyapunov function V_ϕ can certify that a closed invariant set $\mathcal{B}$ containing $\mathcal{A}$ and satisfying $\mu(\mathcal{B} \setminus \mathcal{A}) < \gamma_2$ is UAS. Furthermore, the set K is contained in the ROA of $\mathcal{B}$.*

Proof. By the converse Lyapunov theorem [17], there exists a function V satisfying the Lyapunov conditions on $\mathcal{I}$. Lemma 2 states that there exists a compact set W such that $\mu(\mathcal{I} \setminus W) < \gamma/2$. Since unions preserve compactness we can suppose without loss of generality that W contains the closed ball of radius r for $r > 0$ sufficiently small, denoted as B_r, which lies in the interior of $\mathcal{I}$. By virtue of $\mathcal{I}$ being a forward invariant set contained in the region of attraction, for autonomous systems, asymptotic stability is equivalent to uniform attractive stability, so in particular there exists a time $T > 0$ such that all solutions starting in W will enter $\mathcal{A}_\rho := \{x \in B_r : V(x) \leq \rho\}$ for any $\rho \geq 0$ without leaving $\mathcal{I}$. The continuity of measure and the Lyapunov condition $V(0) = 0$ implies there exists a constant $\rho_0 > 0$ such that $\mu(\mathcal{A}_{\rho_0}) < \gamma/2$ since $\cap_{\rho>0}\mathcal{A}_\rho = \{0\}$ and this is a set of measure zero. For ease of notation, we simply refer to $\mathcal{A}_{\rho_0}$ as $\mathcal{A}$. Let $T \geq 0$ be the time such that all solutions starting in W enter $\mathcal{A}$. By Lemma 1, the reachable set $R^{0\leq t\leq T}(W)$ is compact and satisfies $\mu(\mathcal{I} \setminus R^{0\leq t\leq T}(W)) < \gamma/2$. Denote

$$K := R^{0\leq t\leq T}(W) \cup \mathcal{A}.$$

We see that K which contains the origin is compact and forward invariant. Similarly, by the continuity of of V and the continuity of measure, it follows that $\cap_{\rho>\rho_0}\mathcal{A}_\rho = \mathcal{A}$ and this implies there exists a level set $\mathcal{A}_{\rho_1}$ such that $\mu(\mathcal{A}_{\rho_1} \setminus \mathcal{A}) < \gamma_2$.

By the continuity of V there exists a constant $\delta > 0$ such that $V > \delta$ and $\nabla_f V(x) < -\delta$ on $K \setminus \mathcal{A}$. We can suppose that $\delta < \rho_1 - \rho_0$ in the inequality above. By Theorem 3, there exists a neural network approximation of V denoted V_ϕ satisfying the pointwise bounds $\|V_\phi - V\|_\infty < \delta/2$ and $\|\nabla_f V - \nabla_f V_\phi\|_\infty < \delta/2$ on $K \setminus \mathcal{A}$. This proves that V_ϕ satisfies the Lyapunov conditions on $K \setminus \mathcal{A}$. To summarize we have established that

$$V_\phi(x) > \delta/2 \; \forall x \in K \setminus \mathcal{A},$$

and

$$\nabla_f V_\phi(x) < -\delta/2; \;\; \forall x \in K \setminus \mathcal{A}.$$

By this pointwise bound $\|V_\phi - V\|_\infty < \delta/2$ it follows that

$$\mathcal{A} \subset \mathcal{B} := \{x \in B_r : V_\phi \leq \rho_0 + \delta_2\} \subset \mathcal{A}_{\rho_1}.$$

This proves that $\mu(\mathcal{B} \setminus \mathcal{A}) < \gamma_2$. Now we show that V_ϕ verifies that the set $\mathcal{B}$ is uniformly asymptotically stable.

(**Uniform Stability**) Given $\epsilon > 0$ as per the definition of uniform stability. Denote

$$B_\epsilon(\mathcal{B}) := \cup_{x\in\mathcal{B}} B_\epsilon(x).$$

Without loss of generality by taking $\epsilon < r$ we can assume that $B_\epsilon(\mathcal{B}) \subset K$. Choose $c > 0$ such that

$$0 < c < \min_{\|x\|_{\mathcal{B}}=\epsilon} V_\phi(x)$$

holds. Then by a contradiction argument the set

$$\Omega^c := \{x \in B_\epsilon(\mathcal{B}) : V_\phi(x) \leq c\}$$

is contained in the interior of $B_\epsilon(\mathcal{B})$. By the continuity of V_ϕ and compactness of $\mathcal{B}$, V_ϕ is uniformly continuous on $\mathcal{B}$. Thus, there exists $0 < \delta_\epsilon < \epsilon$ such that

$$\|x_0 - x\| < \delta_\epsilon \implies |V(x) - V(x_0)| < c - \rho_0/2 \text{ for all } x_0 \in \mathcal{B}.$$

In particular, this prove that $B_{\delta_\epsilon}(\mathcal{B}) \subset \Omega^c \subset B_\epsilon(\mathcal{B})$. A standard argument shows that the set Ω^c is forward invariant and hence for all $x_0 \in B_{\delta_\epsilon}(\mathcal{B})$ this implies that $x(t) \in \Omega^c$ for all $t \geq 0$ and proves uniform stability.

(**Uniform Attractivity**). Given that K is a compact positive invariant set that contains $\mathcal{A}$ we claim that there exists some time $T > 0$ for which the solution enters $\mathcal{A}$. Indeed, suppose otherwise this means that $x(t) \in K \setminus \mathcal{A}$ for all $t \geq 0$. By Lemma 3 we get that $\nabla_f V_\phi(x(t)) < -\delta/2$ for all $t \geq 0$ on $K \setminus \mathcal{A}$. It follows that

$$V(x(t)) = V(x(0)) + \int_0^t \nabla_f V_\phi(x(\tau))d\tau \leq V(x(0)) - \delta t/2.$$

As the right hand side will eventually become negative, this is a contradiction to the $V_\phi > 0$ on $K \setminus \mathcal{A}$. To conclude, note that the set $\mathcal{A}$ is forward invariant which implies that $\|x(t)\|_{\mathcal{A}} = 0$ for all $t \geq T$. In particular, as $\mathcal{A}$ is contained in $\mathcal{B}$ this proves the uniform attractivity of $\mathcal{B}$. □

Suppose further that $\mathcal{I}$ is the ROA of the system (3.2). If the Lyapunov function V is *radially unbounded*, this means that $V(x) \to \infty$ when $x \to \delta\mathcal{I}$ (the boundary of $\mathcal{I}$), then the level sets of V approach the ROA. If the origin is UAS for system (3.2), then by the converse Lyapunov theorem, it follows that $V(x)$ is radially unbounded. We show that the neural Lyapunov function inherits a similar property where the level sets approach K.

Theorem 5. *In addition to the assumptions of Theorem 4, suppose that $\mathcal{I}$ is the region of attraction which is bounded. Set $W^c := \{x \in K : V_\phi(x) \leq c\}$. Then, for any sequence $k_n \to \infty$, $\cup_{n\in\mathbb{N}} W^{k_n} = K$.*

Proof. Without loss of generality suppose that k_n is an increasing sequence and $\epsilon < k_1$. Again, define $V^c = \{x \in \mathcal{D} : V(x) \leq c\}$. Note that if $x \in K$ satisfies $V_\phi(x) \leq c$, then $V(x) \leq c + \epsilon$. Therefore, $W^{k_n} \subset V^{k_n+\epsilon} \cap K$. A similar argument shows that $V^{k_n-\epsilon} \cap K \subset W^{k_n} \subset V^{k_n+\epsilon} \cap K$. Since $K \subset \mathcal{D}$ this implies that $\cup_{n\in\mathbb{N}} V^{k_n-\epsilon} \cap K = K \cap \cup_{n\in\mathbb{N}} V^{k_n-\epsilon} = K$. Therefore, $\cup_{n\in\mathbb{N}} W^{k_n} = K$. □

3.2 Learn and Stabilize Dynamics with Neural Lyapunov Functions

In this section we present a learning framework to simultaneously stabilize an unknown nonlinear system with a neural controller and learn a neural Lyapunov function to certify a region of attraction (ROA) for the closed-loop system. We build upon the framework in [7], but address the more challenging question of stabilizing a nonlinear system with unknown dynamics and offering formal guarantees. The theoretical results in the previous section support the proposed method and the framework in [7]. The method involves two shallow neural networks. The first one learns the system dynamics from data and the second one learns a Lyapunov function for the learned dynamics along with its corresponding nonlinear controller. The full learning framework involving the first neural network describe in Section 3.2.1 and the second neural network described in 3.2 can be found in Fig. 3.1 and the pseudocode in Algorithm 1.

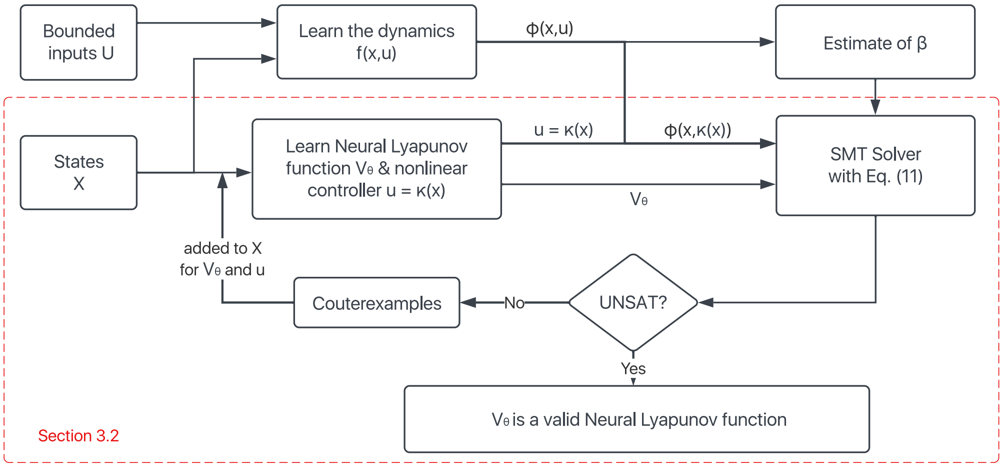


Figure 3.1: The schematic diagram of the proposed algorithm.

3.2.1 Learning Unknown Dynamics

Theorem 2 guarantees the existence of a neural network that approximates the right hand side $f(x,u)$. The first neural network which we denote as $\phi(x,u)$ is a one-hidden-layer feedforward neural network equipped with mean square loss. The activation function is tanh as the activation functions need to be C^1 to calculate its Lie derivatives. Note that the activation function is not applied to the last layer of the neural network as $f(x,u)$ need not be bounded. The training data of x and u are sampled uniformly and independently over their respective spaces. The format of the controller is

$$u = \kappa(x) = C\sigma(kx + b), \tag{3.3}$$

where C is a constant matrix, determined by the saturation property of the controller in real systems, which defines the bounds of $\mathcal{U}$, k and b are the weight matrix and bias vector, respectively, which are initialized with an LQR controller based on the linearized system $\dot{x} = Ax + Bu$. The bias b is chosen such that $f(0, C\sigma(b)) = 0$, i.e., the origin is an equilibrium point for the closed-loop system (3.2). Note that the bias vector b is not updated in the learning process.

3.2.2 Neural Lyapunov Function and Nonlinear Controller

Once the training process has identified the dynamics ϕ, the second neural network simultaneously learns a nonlinear controller and a neural Lyapunov function to certify stability about the origin for the learned dynamics. The detailed structure of the neural network is detailed in Fig. 3.2 [7]. Note that since we are learning a Lyapunov function using the learned dynamics there is no guarantee that this function will certify stability around the origin for the actual dynamics. However, since Lyapunov functions involve pointwise bounds, we introduce an external parameter β that functions as a margin of error and show that the learned Lyapunov function is indeed a Lyapunov function for the actual dynamics. Thus, to account for approximation errors introduced by learning the neural network, the following stricter conditions should be satisfied instead:

$$V(0) = 0, \text{ and }, \forall x \in \mathcal{D}\backslash\{0\}, V(x) > 0 \text{ and } \nabla_{\phi} V(x) < -\beta. \tag{3.4}$$

Here β is a positive real number, which can be determined by . Since computed functions can not become arbitrarily large on $\mathcal{D}$, there exists $M > 0$ be such that $\|\frac{\partial V}{\partial x}\| < M$ for all $x \in \mathcal{D}$. Let δ be the covering number for the space of samples, that is for every unsampled

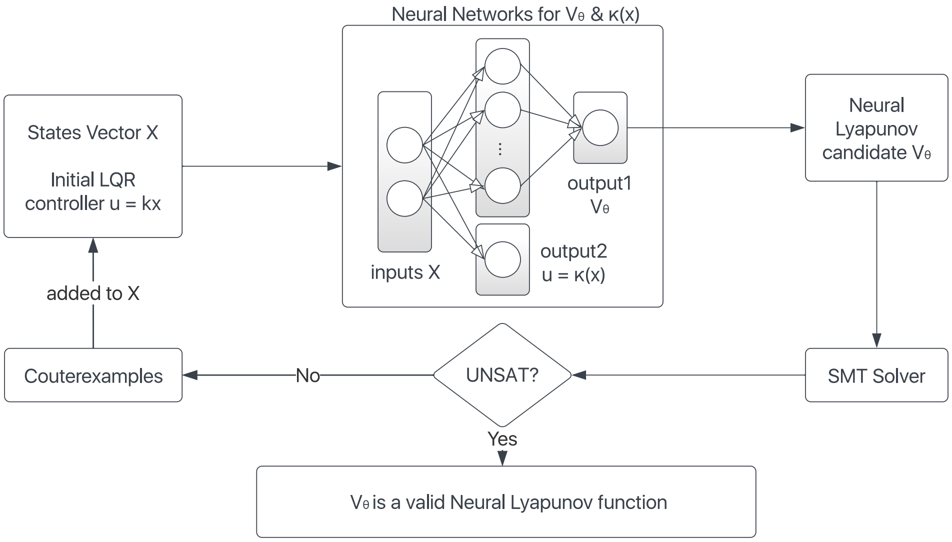

Neural Networks for Vθ & κ(x)
States Vector X
Initial LQR controller u = kx
inputs X
output1 Vθ
output2 u = κ(x)
Neural Lyapunov candidate Vθ
SMT Solver
UNSAT?
No
Yes
Couterexamples
added to X
Vθ is a valid Neural Lyapunov function

Figure 3.2: Algorithmic structure of learning a neural Lyapunov function and the corresponding nonlinear controller with a 1-hidden layer neural network and an SMT solver.

pair of states and input (x, u) there exists a sample point (y, v) used in training and testing the neural network such that

$$\|(x, u) - (y, v)\| \leq \delta. \tag{3.5}$$

Denoting α as the maximum of the 2-norm loss among all known samples, which can be from either the training dataset or the test dataset, the bound on the generalization error can be calculated as

$$\begin{aligned}\|f(x,u) - \phi(x,u)\| &\leq \|f(x,u) - f(y,v)\| + \|f(y,v) - \phi(y,v)\| + \|\phi(y,v) - \phi(x,u)\| \\ &\leq K_f\delta + \alpha + K_\phi\delta < \frac{\beta}{M},\end{aligned} \tag{3.6}$$

where f and ϕ are Lipschitz continuous with Lipschitz constants K_f, K_ϕ, respectively and $\beta > 0$ is some sufficiently large constant. Then, the choice of β to satisfying (3.6) implies that

$$\nabla_f V(x) - \nabla_\phi V(x) \leq \|\frac{\partial V}{\partial x}\|\|f(x,\kappa(x)) - \phi(x,\kappa(x))\| < M\frac{\beta}{M} = \beta. \tag{3.7}$$

In view of (3.4) and (3.7), we have

$$\nabla_f V(x) < \nabla_\phi V(x) + \beta < -\beta + \beta = 0, \quad \forall x \in \mathcal{D}\backslash\{0\}, \tag{3.8}$$

which guarantees that the actual dynamics is also stable with the obtained neural Lyapunov function. Here, we calculate K_ϕ by using LipSDP-network developed in [?], and M can be determined by checking the inequality with an SMT solver. An initial guess of β is needed according to some prior knowledge, and it will be re-computed in each epoch with (3.6).

A valid neural Lyapunov function can be guaranteed by the SMT solver with all the Lyapunov conditions written as falsification constraints in the form of first-order logic formula over reals [7]:

$$\Phi_\varepsilon(x) := \left(\sum_{i=1}^{n} x_i^2 \geq \varepsilon\right) \wedge \left(V(x) \leq 0 \vee \nabla_\phi V(x) \geq -\beta\right), \tag{3.9}$$

where ε is a numerical error parameter, which is explicitly introduced for controlling numerical sensitivity around the origin. If the SMT solver returns either `UNSAT` this means that the falsification constraint is guaranteed not to have any solutions and confirms all the Lyapunov conditions are met. If the SMT solver returns δ-`SAT`, this means there exists at least one counterexample under the δ-weakening condition [?] that satisfies the falsification conditions.

We use θ to denote the parameter vectors for a neural Lyapunov function candidate V_θ. The parameters θ and k are found by minimizing the following cost function, which is a modification of the so-called *empirical Lyapunov risk* in [7] by adding one more term $\|\frac{\partial V}{\partial x}\|$, as we need β to be small as well:

$$L(\theta, k) = \frac{1}{N}\sum_{i=1}^{N}\Big(C_1 \max\left(-V_\theta\left(x_i\right), 0\right) + C_2 \max\left(0, \nabla_\phi V_\theta\left(x_i\right)\right)\Big) + C_3 V_\theta^2(0) + C_4 \|\frac{\partial V}{\partial x}\|, \tag{3.10}$$

where C_1, C_2, C_3 and C_4 are tunable constants. The cost function can be regarded as the positive penalty of any violation of the Lyapunov conditions in (2.3) and (2.4). Note that the ROA can also be maximized by adding an L_2-like cost term to the *Lyapunov risk* with $L(\theta, k) + \frac{1}{N}\sum_{i=1}^{N}\|x_i\|_2 - \alpha V_\theta\left(x_i\right)$, where α is a tunable parameter, as shown in the original paper [7].

Our analysis provides stability guarantees for the unknown system by rigorously quantifying the errors using Lipschitz constants of the unknown dynamics and its neural approximation. To this end, we need a theoretical guarantee that extends the universal approximation theorem, stating that we can approximate the Lipschitz constants and function values by a neural network to an arbitrary precision.

Algorithm 1 Neural Lyapunov Control with Unknown Dynamics

```
function LearningDynamics($X_{dyn}, U_{(bdd)}$)
    Set learning rate ($\gamma$), input dimension ($n + m$), output dimension ($n$)
    repeat
        $f \leftarrow \mathrm{NN}_\phi(x)$                    ▷ Output of forward pass of neural network
        Compute MSE $L(f, \phi)$
        $\phi \leftarrow \phi - \gamma \nabla_\phi L(f, \phi)$                    ▷ Updates Weights using SGD
    until convergence
    return $\phi$
end function
function LearningLyapunov($X_{Lyp}, f_\phi, k^{lqr}$)
    Set learning rate ($\alpha$), input dimension ($n$), output dimension (1)
    Initialize feedback controller u to LQR solution $k^{lqr}$
    repeat
        $V_\theta(x), u(x) \leftarrow NN_{\theta,u}(x)$                    ▷ Output of forward pass of neural network
        $\nabla_\phi V(x) \leftarrow \sum_{i=1}^{D_{in}} \frac{\partial V}{\partial x_i}[\phi]_i(x)$
        Compute Lyapunov risk $L(\theta, k)$
        $\theta \leftarrow \theta - \alpha \nabla_\theta L(\theta, k)$
        $k \leftarrow k - \alpha \nabla_k L(\theta, k)$                    ▷ Updates Weights using SGD
    until convergence
    return $V_\phi, u$
end function
function Falsification($f_\phi, u, V_\theta, \varepsilon, \delta, \beta$)
    Encode conditions from (3.9)
    Use SMT solver with $\delta$ to verify the conditions
    return satisfiability
end function
function Main( )
    input initial guess of bound ($\beta$), parameters of LQR ($k^{lqr}$), radius ($\varepsilon$), precision
  ($\delta$), sampled states $X$, sampled inputs $U$
    $\phi \leftarrow$ LearningDynamics($X_{dyn}, U_{(bdd)}$)
    while Satisfiable do
        Add counterexamples to $X$
        $V_\phi, u \leftarrow$ Learning-Lyapunov($X_{Lyp}, \phi, k^{lqr}$)
        update $\beta$ according to (3.6)
        CE $\leftarrow$ Falsification($\phi, u, V_\theta, \varepsilon, \delta, \beta$)
    end while
end function
```

Theorem 6. *Suppose that $K \subset \mathbb{R}^n$ is a compact set.*
(a) If $f : K \to \mathbb{R}^m$ is L-Lipschitz in the uniform norm, i.e.

$$\|f(x) - f(y)\|_\infty \leq L\|x - y\|_\infty, \tag{3.11}$$

then for every $\epsilon > 0$ there exists a neural network of the form $\phi(x) = C(\sigma \circ (Ax + b))$ for $\sigma \in C^1(\mathbb{R})$ and not a polynomial, $A \in \mathbb{R}^{k\times m}, b \in \mathbb{R}^k$ and $C \in \mathbb{R}^{k\times n}$ for some $k \in \mathbb{N}$ such that $\sup_{x\in K} |f(x) - \phi(x)| < \epsilon$ and ϕ has a Lipschitz constant of $L + \epsilon$ in the same norms as (3.11).
(b) If $f : K \to \mathbb{R}^m$ is L-Lipschitz in the two norm, i.e.

$$\|f(x) - f(y)\|_\infty \leq L\|x - y\|_2, \tag{3.12}$$

then for every $\epsilon > 0$ there exists a neural network ϕ of the same form such that $\sup_{x\in K} |f(x) - \phi(x)| < \epsilon$ and ϕ has a Lipschitz constant of $L+\epsilon\left(\frac{\sqrt{n+n/\epsilon}}{2} + L\right)$ in the same norms as (3.12).

The idea of the proof is to first approximate f by a smooth function F and then approximate F by a neural network ϕ. Since the techniques used in this proof are different from those of this chapter, the details can be found in the appendix.

Remark 4. *The equivalence of norms gives an upper bound on the Lipschitz constant for all norms.*

3.2.3 Asymptotic Stability Guarantees of Unknown Nonlinear Systems

With the theoretical guarantee of learning a neural Lyapunov function established above, we show that the neural network trained with the learned dynamics is robust, that is this neural network also satisfies the Lyapunov conditions with respect to the actual dynamics f. Since the SMT solver verifies the Lyapunov conditions outside of some ϵ-ball which is not necessarily forward invariant, the following technical assumption helps bridge this gap. The assumption is mild, because for the nonlinear system to be stabilizable, it is reasonable to assume that it has a stabilizable linearization. A linear system $\dot{x} = Ax + Bu$ is said to be *stabilizable* if there exists a matrix $\mathcal{K}$ such that $A + B\mathcal{K}$ is Hurwitz, i.e., all eigenvalues of $A + B\mathcal{K}$ have negative real part. If (A, B) is stabilizable, then the gain matrix $\mathcal{K}$ can be obtained by an LQR controller.

Assumption 3 (ROA of LQR Controller)**.** *Suppose that the linearized model $\dot{x} = Ax + Bu$ from Assumption 2 is stabilizable. Consequently, an LQR controller and a quadratic Lyapunov function can be found such that the origin is UAS for the closed-loop system* (3.2)*. Furthermore, we assume that the set B_ε which is not verified by a SMT solver lies in the interior of a ROA of the closed-loop system, provided by the quadratic Lyapunov function.*

Since this ε-ball is small, we further assume that the level sets of the Lyapunov function are contained in the ROA of the LQR controller.

Assumption 4 (Controlled Level Sets)**.** *Denote $B_\varepsilon := \{x : \|x\| \le \varepsilon\}$. Let V be a continuously differentiable function satisfying the Lyapunov conditions on $\mathcal{D} \setminus B_\varepsilon$. Suppose that there exists constants $0 < c_1 < c_2$ such that the following chain of inclusions holds*

$$\{x \in \mathcal{D} : V(x) \le c_1\} \subset B_\varepsilon \subset \{x \in \mathcal{D} : V(x) \le c_2\}, \tag{3.13}$$

and $\{x \in \mathcal{D} : V(x) \le c_2\}$ lies in the interior of the ROA of the closed loop system provided by the quadratic Lyapunov function.

Theorem 7. *Let ϕ be the approximated dynamics of right-hand side of the closed-loop system (3.2) trained by the first neural network. There exists a neural Lyapunov function V which is learned using ϕ and verified by an SMT solver that satisfies the Lyapunov conditions with respect to the actual dynamics f. Furthermore, if the system satisfies Assumption 3 and V satisfies Assumption 4, then the origin is UAS for the closed-loop system* (3.2)*.*

Proof. Fix $\beta > 0$ and let $M > 0$ be chosen such that $\|\frac{\partial V}{\partial x}\| < M$. As V is learned using the learned dynamics ϕ, V satisfies $V \ge 0$ and $-\nabla_\phi V(x) < -\beta$ on $\mathcal{D} \setminus B_\varepsilon$. To certify that V satisfies the Lyapunov conditions on $\mathcal{D} \setminus B_\varepsilon$, it suffices to verify that $\nabla_f V < 0$. By the universal approximation theorem, there exists a neural network ϕ approximating f such that $\|f(x, \kappa(x) - \phi(x, \kappa(x)\|_\infty < \frac{\beta}{M}$ on the $\mathcal{D} \setminus \{\|x\| < \varepsilon\}$. As in (3.8), the following holds

$$\nabla_f V(x) < \nabla_\phi V(x) + \beta < -\beta + \beta = 0, \quad \forall x \in \mathcal{D}\backslash\{0\}. \tag{3.14}$$

Therefore, V satisfies the neural Lyapunov conditions on $\mathcal{D} \setminus B_\varepsilon$.

(**Uniform Stability**). The uniform stability property follows from the Assumption 3 as the quadratic Lyapunov function guarantees uniform stability at the origin.

(**Uniform Attractivity**). We show that under these assumptions, the neural Lyapunov function is able to verify uniform attractivity. As uniform attractivity is equivalent to attractivity for autonomous systems, it suffices to verify that any level set of V, denoted V^c, which contains B_ε is a ROA for this dynamical system. By a similar argument to the

Uniform Stability part of Theorem 4, any trajectory starting in V^c must eventually enter B_ε. Since B_ε is contained in the ROA of the closed loop system provided by the quadratic Lyapunov function, it follows that $\|x(t)\| \to 0$ as $t \to \infty$. □

This theorem proves that if the dynamics are approximated to sufficient precision then the neural Lyapunov function satisfies the Lyapunov conditions on $\mathcal{D} \setminus B_\varepsilon$ for the actual dynamics. Furthermore, if the level sets of the neural Lyapunov function are sufficiently well behaved and the set B_ε excluded from SMT verification is small then this learning framework certifies that the origin is UAS for the actual system (3.2).

3.3 Experiments

In this section we demonstrate the ability of the learning framework in simultaneously learning a non-linear controller and a Lyapunov function that certificies stability about the origin. All examples are carried out using two neural networks, each with one hidden layer. For learning the dynamics, the number of neurons in the hidden layer varies from 100 to 200 without an output layer activation function as stated before, and we call this neural network FNN for convenience. However, for learning the neural Lyapunov function, there are six neurons in the hidden layer for all the experiments, and we name this neural network VNN in short. Regarding other parameters, we use the Adam optimizer for both FNN and VNN, and we use dReal as the SMT solver, setting δ as 0.01 for all experiments. The learned dynamics is in the format of $\phi = W_2 \tanh(W_1 X + B_1) + B_2$ where $X = [x, u]$. In VNN, the valid neural Lyapunov function is of the following form: $V_\theta = \tanh(W_2 \tanh(W_1 x + B_1) + B_2)$. In all experiments since we have access to the non-linear dynamics, we test the neural Lyapunov function on the actual dynamics and observe that these satisfy the Lyapunov conditions on the valid region. All the training of FNN is performed on Google Colab with a 16GB GPU, and VNN training is done on a 3 GHz 6-Core Intel Core i5. The code is written by my colleague and open sourced at https://github.com/RuikunZhou/Unknown_Neural_Lyapunov.

3.3.1 Van der Pol Oscillator

As a starting point, we test the proposed algorithm on the nonlinear system without input u first to show its effectiveness in learning unknown dynamical systems and finding the valid neural Lyapunov function. Van der Pol oscillator is a well-known nonlinear system with stable oscillations as a limit cycle. The area within the limit cycle is the non-convex ROA, as shown in the appendix. The states equations of Van der Pol oscillator are:

$$\begin{aligned}\dot{x}_1 &= -x_2 \\ \dot{x}_2 &= x_1 + \left(x_1^2 - 1\right) x_2.\end{aligned} \tag{3.15}$$

Correspondingly, the phase plot and the limit cycle of this nonlinear system are shown in Figure 3.3 [17]. According to the algorithm described in Section 1, we learn the two nonlinear dynamical equations with 100 hidden neurons. Clearly, we can write $x = [x_1 \quad x_2]^T$, and we use 100 hidden neurons to learn the dynamics in FNN. With the learned dynamics, the weights and biases matrices of obtained neural Lyapunov function in VNN are:

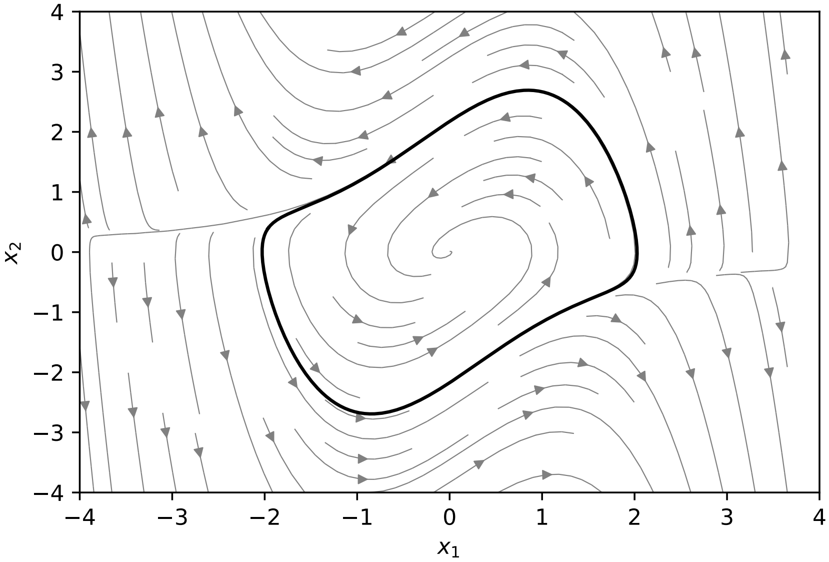


Figure 3.3: Phase space plot and the limit cycle (bold black line) of Val Der Pol oscillator without controller, where the area within the bold black curve forms the actual ROA.

$$W_1 = \begin{bmatrix} -1.82994 & -0.70762 & 3.35979 & -6.42827 & -1.14237 & 0.39034 \\ 1.30866 & 0.57501 & 0.27398 & 0.32546 & -1.16843 & -0.03503 \end{bmatrix}^T,$$

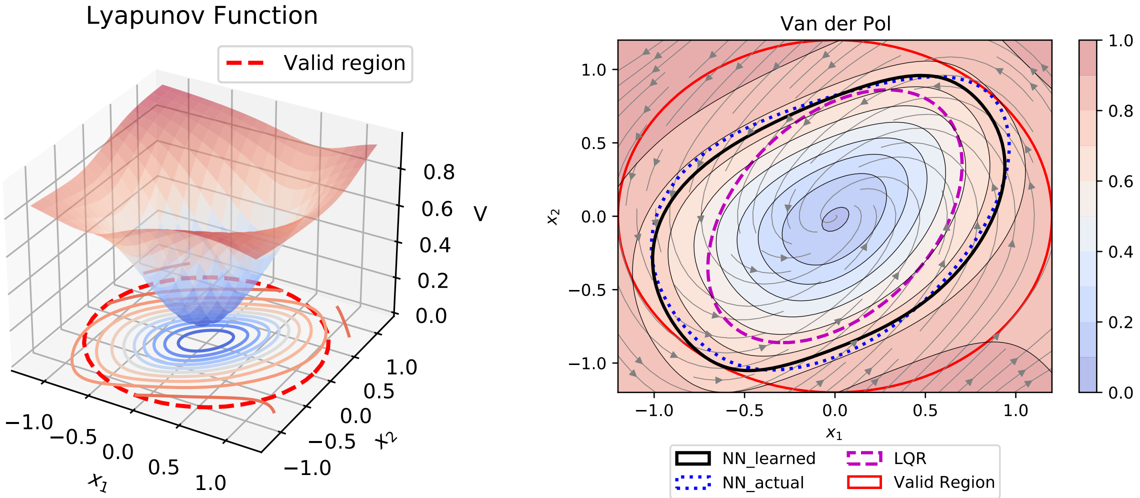
$$W_2 = \begin{bmatrix} -1.32270 & -0.73489 & 1.87897 & 0.89612 & 1.65451 & 1.17499 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} -2.30191 & 0.38658 & 0.47604 & 0.83902 & 0.87791 & 1.18262 \end{bmatrix} \text{ and } B_2 = [0.62172].$$

To ensure an accurate model, we use 9 million data points sampled on $(-1.5, 1.5) \times (-1.5, 1.5)$, and the learning rate of the training process varies from 0.1 to 1e-5 to acquire a small enough α. With the learned dynamics, we aim to find a valid Lyapunov function over the domain $\|x\|_2 < 1.2$, and the obtained neural Lyapunov function is shown in Fig. 3.4a. The corresponding ROA estimate can be found in Fig. 3.4b. In comparison, the ROAs found by the neural Lyapunov function and the classical LQR techniques in [17] using actual dynamics are also plotted, as the blue and magenta ellipses respectively. The phase portrait of the system is given as the grey curves with small arrows. It is obvious that the neural Lyapunov function after tuning obtains a larger ROA. We also have a comparable verified ROA for the system with the one obtained with actual dynamics based on the same neural Lyapunov approach, both larger than the LQR case. The values of the parameters can be found in Table 3.1.

Table 3.1: Parameters in Van der Pol Oscillator case

K_f	K_ϕ	δ	α	$\|\frac{\partial V}{\partial x}\|$	β	ε
3.4599	5.197	5e-4	8.5e-3	1.249	0.02	0.2



(a) Neural Lyapunov function

(b) ROA comparison with LQR

Figure 3.4: Neural Lyapunov function and the corresponding estimated ROA for Van der Pol oscillator.

3.3.2 Unicycle Path Following

The path following task is a typical stabilization problem for a nonlinear system. Here, we consider path tracking control using a kinematic unicycle with error dynamics [5]. In this case, we have two state variables, the angle error θ_e and the distance error d_e, and the dynamics of this system can be written as:

$$\begin{aligned} \dot{s} &= \frac{v\cos(\theta_e)}{1 - d_e\kappa(s)}, \\ \dot{d}_e &= v\sin(\theta_e), \\ \dot{\theta}_e &= \omega - \frac{v\kappa(s)\cos(\theta_e)}{1 - d_e\kappa(s)}. \end{aligned} \tag{3.16}$$

Here we assume the target path is a unit circle $\kappa(s) = 1$ and take ω as the input u with $x = [d_e \quad \theta_e]^T$, consequently the dynamical system is of the format $\dot{x} = f(x, u)$. Similarly, after obtaining the learned dynamics $\phi(x, u)$ with 200 hidden neurons, the weights and biases matrices of V_θ for this experiment are recorded below.

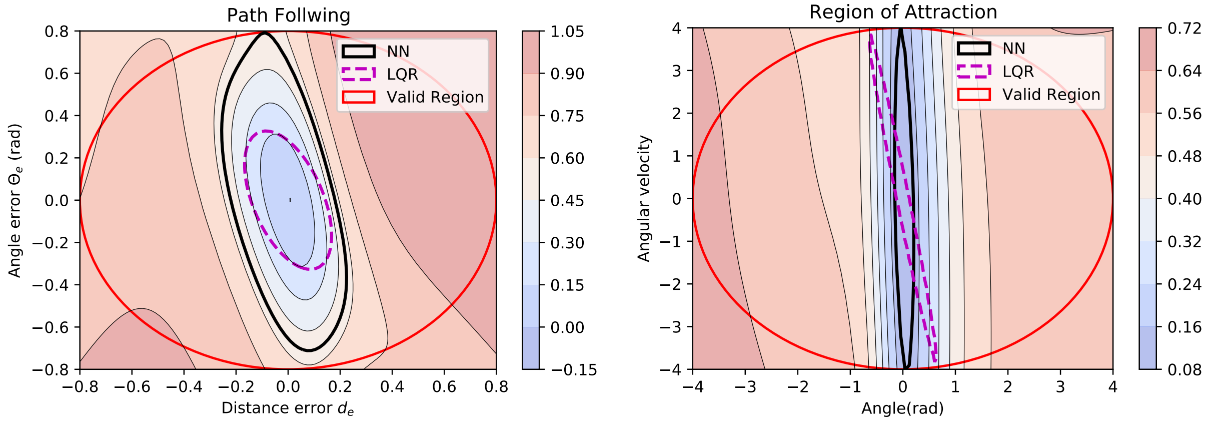
$$W_1 = \begin{bmatrix} -2.13787 & -0.02771 & 2.83659 & -3.33855 & 0.61321 & 4.98050 \\ 1.07949 & -0.25036 & 0.69794 & -2.23639 & -1.62861 & 0.11680 \end{bmatrix}^T,$$
$$W_2 = \begin{bmatrix} -1.23695 & 1.08396 & -2.13833 & -0.76877 & -0.84737 & 1.47562 \end{bmatrix},$$
$$B_1 = \begin{bmatrix} -1.90726 & 0.87544 & 0.18892 & 0.73855 & 1.09844 & -0.79774 \end{bmatrix} \text{ and } B_2 = [0.59095],$$

and the nonlinear controller function is $u = 5\tanh(-5.95539d_e - 4.034226\theta_e + 0.19740)$With the learned dynamics ϕ, a neural Lyapunov function can be learned on the valid region $\|x\|_2 \leq 0.8$, and the neural controller is set as $u = 5\tanh(kx + b)$. The ROA comparison with the LQR method can be found in Fig. 3.5a. Apparently, the neural network method yields a larger estimated ROA, compared to the classical LQR approach in which the level set is determined by considering some relaxation of the largest reasonable range of linearization for practical systems under a small angles assumption, given the fact that the actual dynamics is unknown.

The parameters for this valid neural Lyapunov function and the learned dynamics in this case are listed in Table 3.2. The Lipschitz constant K_f is computed by using the bound $\|J_f\|_2 \leq \sqrt{m}\|J_f\|_\infty$, where J_f is the Jacobian matrix of f and m is the number of rows of J_f. Note that α can be the maximum of the 2-norm loss over a test dataset as stated in Section 3.2, since what we need here is the discrepancies between the actual value and the approximated value of some known samples. In this regard, we can train FNN with fewer data samples, which is more computationally efficient. Then on top of that, a much larger dataset uniformly sampled over state and input spaces is used to calculate α, which contributes to a smaller δ. We implement the same approach on the inverted pendulum case as well.

Table 3.2: Parameters in unicycle path following case

K_f	K_ϕ	δ	α	$\|\frac{\partial V}{\partial x}\|$	β	ε
¡ 45	108	1e-4	7e-3	4.43	0.1	0.1



(a) ROA comparison for path following

(b) ROA comparison for inverted pendulum

Figure 3.5: Comparison of obtained ROAs for path following and inverted pendulum.

3.3.3 Inverted Pendulum.

The inverted pendulum is another well-known nonlinear control problem. This system has two state variables θ, $\dot{\theta}$ and one control input u. Here, θ and $\dot{\theta}$ represent the angular position from the inverted position and angular velocity. The system dynamics of inverted pendulum can be described as

$$\ddot{\theta} = \frac{mg\ell\sin(\theta) + u - 0.1\dot{\theta}}{m\ell^2}. \tag{3.17}$$

In this example, the only nonlinear function we need to learn for FNN is (3.17). Therefore, the input $[x \quad u]^T$ is 3-dimensional and the output is 1-dimensional. Using constants $g = 9.81, m = 0.15$ and $\ell = 0.5$, by the same process as the previous experiment, the weights and bias matrices of the neural Lyapunov function for this experiment are listed below, and the corresponding parameters can be found in Table 3.3.

$$W_1 = \begin{bmatrix} 0.03331 & 0.03467 & 2.12564 & -0.39925 & 0.12885 & 0.95375 \\ -0.03113 & -0.01892 & 0.02354 & -0.10678 & -0.32245 & 0.01298 \end{bmatrix}^T,$$

$$W_2 = \begin{bmatrix} -0.33862 & 0.65177 & -0.52607 & 0.23062 & -0.04802 & 0.66825 \end{bmatrix},$$

$$B_1 = \begin{bmatrix} -0.48061 & 0.88048 & 0.86448 & -0.87253 & 0.81866 & -0.26619 \end{bmatrix} \text{ and } B_2 = [0.22032],$$

and the nonlinear controller function is $u = 20\tanh\left(-23.286632\theta - 5.270555\dot{\theta}\right)$ The valid domain is $\|x\|_2 \leq 4$. The similar ROA comparison is shown in Fig. 3.5b, where the LQR approach uses the same function as given in [7]. The details regarding the bounds and β are given in the appendix.

Table 3.3: Parameters in inverted pendulum case

K_f	K_ϕ	δ	α	$\|\frac{\partial V}{\partial x}\|$	β	ε
¡33.214	633.806	5e-5	5e-3	0.51	0.02	0.4

Chapter 4

Koopman Operator

The classical perspective on dynamical systems that dominated this field in the 1900s is the geometric perspective championed by proficient mathematicians such as Poincare. This approach works well for low-dimensional dynamical systems, but it can not be used in the absence of a well-modeled differential equation. Fortunately, the more recent data driven methods only require measurements of the system and subsequently, can perform system identification and can also determine the Lyapunov function for verifying asymptotic stability as described in the previous chapter. In the 1920s an alternative perspective that analyzed the evolution of so called measurement functions through the flow mapping introduced the Koopman operator. This methodology has shown its significance in data-driven dynamical systems with applications to robotics which are typically difficult to model with differential equations [39].

We will start off by clearly defining the notation and terminology. Denote the state space as $\mathcal{X}$ and define the dynamics on it by some iterated mapping $T : \mathcal{X} \to \mathcal{X}$. In this chapter, we are concerned with observables or measurements of the state. These functions can be thought of as data points. To this end, define an observable to be a function $g : \mathcal{X} \to \mathbb{C}$. The (discrete-time) Koopman operator is denoted as the operator mapping $\mathcal{K}_T : \mathcal{F} \to \mathcal{F}$ and can be defined by the following relation

$$[\mathcal{K}_T g](\mathbf{x}) = g(T(\mathbf{x})). \tag{4.1}$$

Working with discrete time dynamics sometimes poses an unneeded restriction as the natural model for many physical processes can be formulated with continuous-time dynamics. Assume that we have the continuous-time dynamical system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}). \tag{4.2}$$

In this setting we have a family of Koopman operators, $\mathcal{K}^t$ with each one being parameterized by time. The family $\{\mathcal{K}^t\}_{t\in\mathbb{R}}$ forms a semi-group and we call this the *Koopman semigroup.* The action of the Koopman semigroup on the observable $g \in \mathcal{F}$ can be defined as

$$[\mathcal{K}^t g](x) = g(\Phi^t(x)). \tag{4.3}$$

Here $\Phi^t(\mathbf{x})$ is the flow map that takes an initial condition $\mathbf{x} \in \mathcal{X}$ and maps it to the solution of the initial value problem (4.3) having initial condition $\mathbf{x}(0) = \mathbf{x}_0$. The *infinitesimal generator* of the Koopman semigroup can be defined as

$$[\mathcal{L}f] := \lim_{t\to 0} \frac{\mathcal{K}^t f - f}{t}, \tag{4.4}$$

where the limit is taken in the strong sense for operator derivatives. We define the *Lie derivative* of g along the vector field given by (3.1) as the scalar function

$$\nabla g \cdot \mathbf{f}(\mathbf{x}(t)).$$

From now on we refer to this scalar function as simply the Lie derivative as the vector field is taken to be implicitly defined. The generator $\mathcal{L}$ is called the Lie operator as the action of this operator is equivalent to taking the Lie derivative of g. By the chain rule, it can be shown that the following holds:

$$\frac{d}{dt} g(\mathbf{x}(t)) = \nabla g \cdot \dot{\mathbf{x}}(t) = \nabla g \cdot \mathbf{f}(\mathbf{x}(t)) \tag{4.5}$$

and equating this with the definition of the infinitesimal generator

$$\frac{d}{dt} g(\mathbf{x}(t)) = \lim_{\tau\to 0} \frac{g(\mathbf{x}(t+\tau)) - g(\mathbf{x}(t))}{\tau} = \mathcal{L}(g(\mathbf{x}(t))) \tag{4.6}$$

results in the following equality

$$\mathcal{L}g = \nabla g \cdot \mathbf{f}. \tag{4.7}$$

4.1 Eigenfunctions and Spectrum of the Eigenvalues

In discrete time, a Koopman eigenfunction $\varphi(\mathbf{x})$ corresponding to an eigenvalue λ satisfies

$$\varphi\left(\mathbf{x}_{k+1}\right) = \mathcal{K}\varphi\left(\mathbf{x}_k\right) = \lambda\varphi\left(\mathbf{x}_k\right) \tag{4.8}$$

and in continuous time, an eigenfunction of the infinitesimal generator $\varphi(\mathbf{x})$ satisfies

$$\frac{d}{dt}\varphi(\mathbf{x}) = \mathcal{L}\varphi(\mathbf{x}) = \mu\varphi(\mathbf{x}). \tag{4.9}$$

In continuous time, each eigenfunction of $\mathcal{K}^t$ is an eigenfunction of the infinitesimal generator, but with a different eigenvalue. Indeed, if we have that $\mathcal{K}^t\varphi = \lambda^t\varphi$, then

$$\mathcal{L}\varphi = \lim_{t\to 0}\frac{\mathcal{K}^t\varphi - \varphi}{t} = \lim_{t\to 0}\frac{\lambda^t - 1}{t}\varphi = \log(\lambda)\varphi, \tag{4.10}$$

by L'Hopital's rule. Therefore, there is a natural correspondence between eigenvalues of $\mathcal{K}^t$ and $\mathcal{L}$ which is given by $\lambda^t = \exp(\mu t)$. This is why we will not make a distinction between Koopman eigenfunctions and eigenfunctions of the infinitesimal generator in this thesis.

Note that the Koopman operator is clearly linear and that by the following result there are infinitely many Koopman eigenfunctions.

Property 1. *The product of Koopman eigenfunctions is a Koopman eigenfunction.*

Proof.

$$\mathcal{K}\left(\varphi_1\varphi_2(\mathbf{x})\right) = \varphi_1\varphi_2(T(\mathbf{x})) = \varphi_1(T(\mathbf{x}))\varphi_2(T(\mathbf{x})) = \lambda_1\lambda_2\varphi_1(\mathbf{x})\varphi_2(\mathbf{x}).$$

□

Note that while a high-dimensional linear approximation simplifies the representation of the dynamics, performing calculations with a high-dimensional matrix (i.e. with matrix multiplication) does not scale linearly in the number of operators so we can look for a representation in terms of *Koopman modes* that will. Although we hae spoken thus far about scalar measurements, in general, we often take multiple measurements of a system, which we will arrange in a vector $\mathbf{g}$:

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_p(\mathbf{x}) \end{bmatrix}.$$

Each of the individual measurements may be expanded in terms of a basis of eigenfunctions $\varphi_j(\mathbf{x})$:

$$g_i(\mathbf{x}) = \sum_{j=1}^{\infty} v_{ij}\varphi_j(\mathbf{x}).$$

Thus, the vector of observables, g, may be similarly expanded:

$$\mathbf{g}(\mathbf{x}) = \begin{bmatrix} g_1(\mathbf{x}) \\ g_2(\mathbf{x}) \\ \vdots \\ g_p(\mathbf{x}) \end{bmatrix} = \sum_{j=1}^{\infty} \varphi_j(\mathbf{x})\mathbf{v}_j, \tag{4.11}$$

where $\mathbf{v}_j$ is known as the j-th Koopman mode associated with the eigenfunction φ_j.

Given the decomposition in (4.11), it is possible to represent the dynamics of the measurements $\mathbf{g}$ as follows:

$$\begin{aligned} \mathbf{g}(\mathbf{x}(t)) = \mathcal{K}^t \mathbf{g}\left(\mathbf{x}_0\right) &= \mathcal{K}^t \sum_{j=1}^{\infty} \varphi_j\left(\mathbf{x}_0\right)\mathbf{v}_j \\ &= \sum_{j=1}^{\infty} \mathcal{K}^t \varphi_j\left(\mathbf{x}_0\right)\mathbf{v}_j \\ &= \sum_{j=1}^{\infty} \lambda_j^t \varphi_j\left(\mathbf{x}_0\right)\mathbf{v}_j. \end{aligned}$$

This sequence of triples $\{(\lambda_j, \varphi_j, \mathbf{v}_j)\}_{j=1}^{\infty}$ is the Koopman mode decomposition and was introduced by Mezić in [23]. Often, it is possible to approximate this expansion as a truncated sum of only a few dominant terms.

4.2 Dynamic Mode Decomposition

For the Koopman theory to be useful in practice, we must be able use the data to learn the Koopman operator. As the Koopman operator is an infinite-dimensional operator, this is not possible, but we can settle for a high dimensional linear representation that approximates the action of the Koopman operator. Since the method has to be computationally tractable, we restrict ourselves to a finite-dimensional subspace of measurement functions $\mathcal{F}_N \subset \mathcal{F}$. A practical interpretation of $\mathcal{F}_N$ is a space of N linearly independent functions that serve as a finite-dimensional basis of $\mathcal{F}$. This section describes exactly this procedure and derives a connection between the *dynamic mode decomposition* (DMD), an algorithm that seeks to obtain a finite-dimensional approximation of the Koopman operator and the restriction of the Koopman operator onto $\mathcal{F}_N$. We follow the procedure outlined in [16], which is defined for discrete time. The advantage of this method over the original formulation of the DMD algorithm introduced by Schmidt in [36] is that the timesteps are not required to be uniform.

The DMD algorithm computes a linear best fit matrix that approximates one forward advancement in time according to the following linear dynamical system

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k. \tag{4.12}$$

Here $\mathbf{x}_k = \mathbf{x}(k\Delta t)$, and Δt denotes a fixed time step that is chosen a priori sufficiently small to resolve the highest frequencies in the dynamics so that the approximation is reasonable. It is further claimed, but never shown in [[10] that the DMD matrix $\mathbf{A}$ is an approximation of the (discrete time) Koopman operator $\mathcal{K}$ restricted to a measurement subspace spanned by direct measurements of the state $\mathbf{x}$. This is true for a generalization of the DMD algorithm called the extended DMD algorithm (eDMD) and is shown in the following section.

The data for the DMD algorithm are pairs of measurements that differ by one time step, i.e. $\{(\mathbf{x}(t_k), \mathbf{x}(t'_k)\}_{k=1}^{m}$ where $t'_k = t_k + \Delta t$. Note that in this algorithm, the times t_k need not be sequential or evenly spaced. These snapshot pairs are then arranged into the following data matrices

$$\begin{aligned} \mathbf{X} &= \begin{bmatrix} | & | & & | \\ \mathbf{x}(t_1) & \mathbf{x}(t_2) & \cdots & \mathbf{x}(t_m) \\ | & | & & | \end{bmatrix}, \\ \mathbf{X}' &= \begin{bmatrix} | & | & & | \\ \mathbf{x}(t'_1) & \mathbf{x}(t'_2) & \cdots & \mathbf{x}(t'_m) \\ | & | & & | \end{bmatrix}. \end{aligned} \tag{4.13}$$

Thus, in matrix notation, (4.12) can be rewritten as

$$\mathbf{X}' \approx \mathbf{A}\mathbf{X}. \tag{4.14}$$

Since (4.14) is typically over-determined and thus, can only be solved approximately, a reasonable approach is to formulate this as an optimization problem

$$\mathbf{A} = \underset{\mathbf{A}}{\operatorname{argmin}} \left\|\mathbf{X}' - \mathbf{A}\mathbf{X}\right\|_F = \mathbf{X}'\mathbf{X}^{\dagger}, \tag{4.15}$$

where $\|\cdot\|_F$ is the Frobenius norm and † denotes the pseudo-inverse. The pseudoinverse of $\mathbf{X}$ may be efficiently solved using the singular value decomposition (SVD). The SVD of $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^*$ where $\mathbf{U}$ and $\mathbf{V}$ are unitary matrices, i.e. $\mathbf{U}^*\mathbf{U} = \mathbf{I}$ and $\mathbf{V}^*\mathbf{V} = \mathbf{I}$. Thus,

$$\mathbf{X}^{\dagger} = \mathbf{V}\mathbf{\Sigma}^{-1}\mathbf{U}^* \tag{4.16}$$

Since the matrix $\mathbf{A}$ has size along the order n^2 we seek a truncated representation which is derived through the leading spectral decomposition. The idea will be to represent the observables in as linear combinations of eigenfunctions of $\mathbf{A}$ so that $\mathbf{A}$ need not be computed. First, observe that since the data matrices $\mathbf{X}$ and $\mathbf{X}'$ have many more rows than columns, say $m << n$, hence the rank of $\mathbf{A}$ which we will denote as r will be an order of magnitude smaller than n. Therefore, for efficiency we project $\mathbf{A}$ onto the first r columns of the matrix $\mathbf{U}$, we denote the matrix formed from the first r columns of $\mathbf{U}$ as $\mathbf{U}_r$ and approximate the pseudo-inverse using the rank-r SVD approximation $\mathbf{X} \approx \mathbf{U}_r \mathbf{\Sigma}_r \mathbf{V}_r^*$. Therefore, the matrix A can be approximated as

$$\begin{aligned} \tilde{\mathbf{A}} &= \mathbf{U}_r^* \mathbf{A} \mathbf{U}_r \\ &= \mathbf{U}_r^* \mathbf{X}' \mathbf{X}^\dagger \mathbf{U}_r \\ &= \mathbf{U}_r^* \mathbf{X}' \mathbf{V}_r \mathbf{\Sigma}_r^{-1} \mathbf{U}_r^* \mathbf{U}_r \\ &= \mathbf{U}_r^* \mathbf{X}' \mathbf{V}_r \mathbf{\Sigma}_r^{-1}. \end{aligned} \tag{4.17}$$

Thus, the spectral decomposition of the matrix A may be approximated from the spectral decomposition of the reduced matrix $\tilde{\mathbf{A}}$ according to

$$\tilde{\mathbf{A}} \mathbf{W} = \mathbf{W} \mathbf{\Lambda}. \tag{4.18}$$

The eigenvectors of $\mathbf{A}$ are the *DMD modes* $\mathbf{\Phi}$, and they are reconstructed using the eigenvectors $\mathbf{W}$ of the reduced system and the time-shifted data matrix $\mathbf{X}'$:

$$\mathbf{\Phi} = \mathbf{X}' \tilde{\mathbf{V}} \tilde{\mathbf{\Sigma}}^{-1} \mathbf{W}. \tag{4.19}$$

Similarly, the *DMD eigenvalues* are the eigenvalues contained in the diagonal matrix Λ. Upon computing the DMD eigenvalues and modes, it is possible to represent future states in a DMD expansion that is analogous to the Koopman mode decomposition

$$\mathbf{x}_k = \sum_{j=1}^{r} \boldsymbol{\phi}_j \lambda_j^{k-1} b_j = \mathbf{\Phi} \mathbf{\Lambda}^{k-1} \mathbf{b}, \tag{4.20}$$

where $\boldsymbol{\phi}_j$ are eigenvectors of $\mathbf{A}$ (DMD modes), λ_j are eigenvalues of $\mathbf{A}$ (DMD eigenvalues) and b_j are the mode amplitudes. Evaluating the Koopman mode amplitudes is equivalent to evaluating for the (initial condition) in the Koopman mode decomposition and is normally evaluated according to

$$\mathbf{b} = \mathbf{\Phi}^\dagger \mathbf{x}_1. \tag{4.21}$$

4.3 Extended DMD and Convergence

While DMD remains a popular algorithm for estimating the Koopman operator, this method struggles with strongly non-linear systems as it is based on linear measurements of state variables. An alternative method introduced in [44] called the extended dynamic mode decomposition (eDMD) takes measurements of the state through non-linear basis functions. Similarly to DMD, for the eDMD algorithm, let $\mathcal{F}$ be the space of observable functions and assume that we are given snapshots of the data X and X' as

$$\begin{aligned}\mathbf{X} &= \left[\begin{array}{cccc} | & | & & | \\ \mathbf{x}(t_1) & \mathbf{x}(t_2) & \cdots & \mathbf{x}(t_m) \\ | & | & & | \end{array}\right], \\ \mathbf{X}' &= \left[\begin{array}{cccc} | & | & & | \\ \mathbf{x}(t_1') & \mathbf{x}(t_2') & \cdots & \mathbf{x}(t_m') \\ | & | & & | \end{array}\right].\end{aligned} \tag{4.22}$$

Then given a set of linearly independent basis functions $f_i \in \mathcal{F}$, define the linear subspace $\mathcal{F}_n \subset \mathcal{F}$ as

$$\mathcal{F}_n := \operatorname{span}\{f_1, \ldots, f_n\}. \tag{4.23}$$

In eDMD, an augmented state $\mathbf{z} \in \mathbb{R}^n$ is formed by

$$\mathbf{z} = \mathbf{f}(\mathbf{x}) = \left[\begin{array}{c} f_1(\mathbf{x}) \\ f_2(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{array}\right], \qquad \mathbf{z}' = \mathbf{f}(\mathbf{x}') = \left[\begin{array}{c} f_1(\mathbf{x}') \\ f_2(\mathbf{x}') \\ \vdots \\ f_n(\mathbf{x}') \end{array}\right]. \tag{4.24}$$

Note that the vector $\mathbf{z}$ may contain the original state if $f_i(\mathbf{x}) = \mathbf{x}$. So we typically have that the dimension of $\mathbf{z}$ is much greater than the dimension of $\mathbf{x}$. Next, construct the data matrices

$$\mathbf{Z} = \left[\begin{array}{cccc} | & | & & | \\ \mathbf{z}_1 & \mathbf{z}_2 & \cdots & \mathbf{z}_m \\ | & | & & | \end{array}\right], \qquad \mathbf{Z}' = \left[\begin{array}{cccc} | & | & & | \\ \mathbf{z}_1' & \mathbf{z}_2' & \cdots & \mathbf{z}_m' \\ | & | & & | \end{array}\right], \tag{4.25}$$

where the sampling need not be uniform in time. The eDMD computes a finite-dimensional approximation of the Koopman operator, $\mathcal{K}_{n,m} : \mathcal{F}_n \to \mathcal{F}_n$, by solving the following least squares problem

$$\min_{A \in \mathbb{C}^{N \times N}} \|A\mathbf{Z} - \mathbf{Z}'\|_F^2 = \min_{A \in \mathbb{C}^{N \times N}} \sum_{i=1}^{m} \|A\mathbf{z}_i - \mathbf{z}_i'\|_2^2. \tag{4.26}$$

Denote the solution to this least squares problem as

$$A_{n,m} = \mathbf{Z}'\mathbf{Z}^{\dagger}. \tag{4.27}$$

We can write the *eDMD operator* $\mathcal{K}_{n,m} : \mathcal{F}_n \to \mathcal{F}_n$ as

$$\mathcal{K}_{n,m}\psi = c_{\psi}^{\mathrm{H}} A_{n,m}\mathbf{f} \tag{4.28}$$

for any $\psi = c_{\psi}^{\mathrm{H}}\mathbf{f}, c_{\psi} \in \mathbb{C}^N$, where H denotes the Hermitian transpose.

For the remainder of this section, we set the space of observables as $L^2(\mu)$ where μ is some positive measure on the state $\mathcal{X}$ (more information on L^2 spaces is given in A. Now, given the data points $x_1, \ldots, x_M$, we define the empirical measure $\hat{\mu}_M$ by

$$\hat{\mu}_M = \frac{1}{M}\sum_{i=1}^{M} \delta_{x_i},$$

where δ_{x_i} is the Dirac measure at x_i. In particular, the integral of a function ϕ with respect to $\hat{\mu}_M$ is given by

$$\int_{\mathcal{X}} \phi(x) d\hat{\mu}_M(x) = \frac{1}{M}\sum_{i=1}^{M} \phi\left(x_i\right).$$

We can now show that under certain conditions computing the eDMD operator from sampled data can be viewed as an L^2 projection. See Appendix A for notation regarding the projection operator in L^2 spaces. First, note that using our notation, (A.1) becomes

$$P_N^{\mu}\phi = \underset{c \in \mathbb{C}^N}{\arg\min} \int_{\mathcal{X}} \left|c^{\mathrm{H}}\mathbf{f} - \phi\right|^2 d\mu. \tag{4.29}$$

Remark 5. *Therefore, the projection with respect to the empirical measure is given by*

$$P_N^{\hat{\mu}_M}\phi = \underset{c \in \mathbb{C}^N}{\arg\min} \sum_{i=1}^{M} \left|c^{\mathrm{H}}\boldsymbol{f}(x_i) - \phi(x_i)\right|^2. \tag{4.30}$$

Since a finite linear combination of L^2 functions is L^2, the subspace $\mathcal{F}_N$ is a closed subspace of both $L_2\left(\hat{\mu}_M\right)$ and $L_2(\mu)$ and hence the projection operators P_N^{μ} and $P_N^{\hat{\mu}_M}$ are well defined.

In the following section we present a proof of the convergence of the eDMD operator to the Koopman operator as presented in [18]. However, we first provide an alternative proof of the following thoerem which has the added benefit of not requiring the assumption that the authors make in [18].

Theorem 8. *Let $\mathcal{F}$ be any space of observables and let $\mathcal{F}_N$ be a subspace spanned by N linearly independent basis functions $f_i \in \mathcal{F}$. Then, the eDMD operator $\mathcal{K}_{m,n} : \mathcal{F}_N \to \mathcal{F}_N$ is the matrix representation of the operator $P_N^{\hat{\mu}_M} \mathcal{K}|_{\mathcal{F}_N}$, where $\mathcal{K}|_{\mathcal{F}_N}$ is the restriction of the Koopman operator to the subspace $\mathcal{F}_N$.*

Proof. By definition it follows that

$$g = \mathbf{a}^T \mathbf{f}, \quad \mathcal{K}_{N,M} g = \mathbf{b}^T \mathbf{f}. \tag{4.31}$$

Since this is clearly a linear transformation it admits a matrix representation which we can define as $\overline{\mathrm{K}}_{N,M} \in \mathbb{R}^{N \times N}$ such that

$$\overline{\mathbf{K}}_{N,M} \mathbf{a} = \mathbf{b}. \tag{4.32}$$

It follows from (4.31) that

$$\mathcal{K}_{N,M} f = \mathcal{K}_{N,M} \left(\mathbf{a}^T \mathbf{f} \right) = \left(\overline{\mathbf{K}}_{N,M} \mathbf{a} \right)^T \mathbf{f}.$$

and, since (4.32) holds for all $\mathbf{a}$, (we can choose $\mathbf{a} = e_i$) to see that

$$\mathcal{K}_{N,M} \mathbf{f}^T = \mathbf{f}^T \overline{\mathbf{K}}_{N,M}.$$

As in (4.30), we have that

$$P_N^{\hat{\mu}_M} \phi = \underset{\tilde{\phi} \in \operatorname{span}\{f_1,\ldots,f_N\}}{\operatorname{argmin}} \sum_{k=1}^{K} \left| \tilde{\phi}\left(\mathbf{x}_k\right) - \phi\left(\mathbf{x}_k\right) \right|^2. \tag{4.33}$$

This corresponds to the least squares solution

$$P_N \phi = \mathbf{f}^T \mathbf{X}_{\mathbf{x}}^{\dagger} \begin{pmatrix} \phi\left(\mathbf{x}_1\right) \\ \vdots \\ \phi\left(\mathbf{x}_K\right) \end{pmatrix}. \tag{4.34}$$

For $\phi = \mathcal{K} f_j$,

$$P_N \left(\mathcal{K} f_j \right) = \mathbf{f}^T \mathbf{X}_{\mathbf{x}}^{\dagger} \begin{pmatrix} \mathcal{K} f_j\left(\mathbf{x}_1\right) \\ \vdots \\ \mathcal{K} f_j\left(\mathbf{x}_K\right) \end{pmatrix} = \mathbf{f}^T \mathbf{X}_{\mathbf{x}}^{\dagger} \begin{pmatrix} f_j\left(\mathbf{y}_1\right) \\ \vdots \\ f_j\left(\mathbf{y}_K\right) \end{pmatrix}. \tag{4.35}$$

Therefore, we have $\mathcal{K}_{N,M} \mathbf{f}^T = \mathbf{f}^T \mathbf{Z}^{\dagger} \mathbf{Z}$ so that we get

$$\overline{\mathbf{K}}_{N,M} = \mathbf{Z}^{\dagger} \mathbf{Z}. \tag{4.36}$$

This proves the claim. □

4.3.1 Convergence of the eDMD Operator

It has been shown that the eigenvectors of the eDMD operator can be used as Lyapunov functions to certify stability for the underlying dynamical system ([22]). Therefore, we wish to understand the convergence of the spectrum under the asymptotic convergence of the eDMD operator to the Koopman operator to develop rigorous guarantees concerning the existence of the Lyapunov function. However, these types of results are rather restrictive as strong notions of spectral convergence can only be formulated in the case of uniform convergence of operators.

Definition 6. *A sequence of bounded linear operators $F_n : V \to W$ is said to converge uniformly to $F : V \to W$ if*

$$\|F_n - F\| \to 0,$$

where $\|\cdot\|$ is any operator norm.

Lemma 3. *Suppose that a sequence of bounded linear operators $F_n : V \to W$ converges uniformly to $F : V \to W$. Then, the respective spectrum converge in the Hausdorff metric, i.e.*

$$\lim_{M\to\infty} \operatorname{dist}\left(\sigma\left(\mathcal{K}_{N,M}\right), \sigma\left(\mathcal{K}_N\right)\right) = 0,$$

where $\sigma(\cdot) \subset \mathbb{C}$ denotes the spectrum of an operator and $\operatorname{dist}(\cdot,\cdot)$ *the Hausdorff metric on subsets of $\mathbb{C}$.*

To discuss the convergence of the eDMD operator, we first make the assumption that the basis functions are linearly independent on all states apart from a set of measure 0. However, as we consider the convergence as the number of samples goes to infinity, we first assume that the measure μ define on $\mathcal{X}$ is a probability measure.

Assumption 5. *Assume that the measure μ defined on $\sigma(\mathcal{X})$, where $\sigma(\mathcal{X})$ is some σ-algebra is a probability measure, that is $\mu(\mathcal{X}) = 1$.*

Assumption 6. *(μ independence) The basis functions $f_1, \ldots, f_N$ are such that*

$$\mu\left\{x \in \mathcal{X} \mid c^{\mathrm{H}} \boldsymbol{f}(x) = 0\right\} = 0$$

for all nonzero $c \in \mathbb{C}^N$.

Remark 6. *Note that for commonly used basis functions like polynomials or trigonometric this assumption holds deterministically.*

It is well known from probability theory that the empirical measure $\hat{\nu}_M$ will converge to ν weakly (weak convergence for probability measures is defined in A). In fact, it is also true that the respective projections converge strongly.

Lemma 4. *If Assumption 6 holds, then for any $\phi \in \mathcal{F}$ we have with probability one*

$$\lim_{M\to\infty} \left\| P_N^{\hat{\mu}_M} \phi - P_N^{\mu} \phi \right\| = 0,$$

where $\|\cdot\|$ is any norm on $\mathcal{F}_N$ (which are all equivalent since $\mathcal{F}_N$ is finite-dimensional).

Remark 7. *Here probability enters the picture as the projection operator $P_N^{\hat{\mu}_M}$ is dependent on the sample points which are sampled according to the distribution μ.*

Since strong convergence is equivalent to uniform convergence in finite-dimensional spaces, we immediately obtain uniform convergence and thus, convergence of the spectrum in the Hausdorff metric.

Theorem 9. *If Assumption 6 holds, then we have with probability one for all $\phi \in \mathcal{F}_N$*

$$\lim_{M\to\infty} \|\mathcal{K}_{N,M}\phi - \mathcal{K}_N\phi\| = 0,$$

where $\|\cdot\|$ is any norm on $\mathcal{F}_N$. In particular

$$\lim_{M\to\infty} \|\mathcal{K}_{N,M} - \mathcal{K}_N\| = 0,$$

where $\|\cdot\|$ is any operator norm and

$$\lim_{M\to\infty} \operatorname{dist}\left(\sigma\left(\mathcal{K}_{N,M}\right), \sigma\left(\mathcal{K}_N\right)\right) = 0,$$

where $\sigma(\cdot) \subset \mathbb{C}$ denotes the spectrum of an operator and $\operatorname{dist}(\cdot,\cdot)$ the Hausdorff metric on subsets of $\mathbb{C}$.

Now, to discuss the convergence of $\mathcal{K}_N$ to the Koopman operator $\mathcal{K}$ we make two further assumptions.

Assumption 7. *The following conditions hold:*
1. The Koopman operator $\mathcal{K} : \mathcal{F} \to \mathcal{F}$ is bounded.
2. The observables $\psi_1, \ldots, \psi_N$ defining $\mathcal{F}_N$ are selected from a given orthonormal basis of $\mathcal{F}$, i.e., $(\psi_i)_{i=1}^{\infty}$ is an orthonormal basis of $\mathcal{F}$.

The following result is true for any separable Hilbert space and therefore holds for $L^2(\nu)$ [12].

Theorem 10. *For an orthonormal sequence $\{e_1, e_2, \ldots\}$ in a separable Hilbert space $\mathcal{H}$, the following statements are equivalent:*
(a) The sequence is an orthonormal basis.
(b) The only vector perpendicular to e_j for all j is zero.
(c) The projection operator $P_n \to I$ in the strong operator sense as $n \to \infty$.

To study the convergence of the spectrum of $\mathcal{K}_N$ to $\mathcal{K}$, we need to redefine the $\mathcal{K}_N$ as this operator is defined on $\mathcal{F}_N$, a finite-dimensional linear space. Thus, the natural extension of $\mathcal{K}_N$ to $\mathcal{F}$ is to precompose with P_N^μ which simply adds a zero to the spectrum of $\mathcal{K}_N$ as all functions orthogonal to $\mathcal{F}_N$ must be mapped to zero under the projection operator. That is, we study the convergence of $\mathcal{K}_N P_N^\mu = P_N^\mu \mathcal{K} P_N^\mu : \mathcal{F} \to \mathcal{F}$ to $\mathcal{K} : \mathcal{F} \to \mathcal{F}$ as $N \to \infty$.

Theorem 11. *If Assumption 7 holds, then the sequence of operators $\mathcal{K}_N P_N^\mu = P_N^\mu \mathcal{K} P_N^\mu$ converges strongly to $\mathcal{K}$ as $N \to \infty$, i.e.,*

$$\lim_{N\to\infty} \int_{\mathcal{X}} \left|\mathcal{K}_N P_N^\mu \phi - \mathcal{K}\phi\right|^2 d\mu = 0$$

for all $\phi \in \mathcal{F}$.

Proof. Let $\phi \in \mathcal{F}$ be given. Then, writing $\phi = P_N^\mu \phi + (I - P_N^\mu)\,\phi$, we have

$$\begin{aligned}\|P_N^\mu \mathcal{K} P_N^\mu \phi - \mathcal{K}\phi\| &= \|(P_N^\mu - I)\,\mathcal{K} P_N^\mu \phi + \mathcal{K}\,(P_N^\mu - I)\,\phi\| \le \|(P_N^\mu - I)\,\mathcal{K} P_N^\mu \phi\| + \|\mathcal{K}\|\,\|(I - P_N^\mu)\,\phi\| \\ &\le \|(P_N^\mu - I)\,\mathcal{K}\phi\| + \|(P_N^\mu - I)\|\,\|\mathcal{K} P_N^\mu \phi - \mathcal{K}\phi\| + \|\mathcal{K}\|\,\|(I - P_N^\mu)\,\phi\| \to 0\end{aligned}$$

by Lemma 2 and by the fact that $\mathcal{K} P_N^\mu \phi \to \mathcal{K}\phi$ since $\mathcal{K}$ is continuous by Assumption 7. □

Since strong convergence is a weaker notion of convergence than uniform convergence, we cannot conclude convergence of the spectrum as in Lemma 3. Accordingly, we can only obtain a weak notion of convergence of the eigenfunctions.

Definition 7. *(Weak Convergence) A sequence of elements $f_i \in \mathcal{F}$ of a Hilbert space $\mathcal{F}$ converges weakly to $f \in \mathcal{F}$, denoted $f_i \xrightarrow{w} f$, if*

$$\lim_{i\to\infty} \langle f_i, g\rangle = \langle f, g\rangle$$

for all $g \in \mathcal{F}$.

Since some results used in the proof of the following theorem are required in for Section 4.4.3, we detail the proof. The following two results can be found in [35].

Theorem 12. *(Banach-Alaoglu theorem: the original version) Let X be a Banach space. Then the unit ball B of X^* is compact in the w^* topology.*

Theorem 13. *(Eberlein-Šmulian theorem) If X is a Banach space and A is a subset of X, then the following statements are equivalent:*

1. *Each sequence of elements of A has a subsequence that is weakly convergent in X.*
2. *Each sequence of elements of A has a weak cluster point in X.*
3. *The weak closure of A is weakly compact.*

Theorem 14. *If Assumption 2 holds and λ_N is a sequence of eigenvalues of $\mathcal{K}_N$ with the associated normalized eigenfunctions $\phi_N \in \mathcal{F}_N, \|\phi_N\| = 1$, then there exists a subsequence $(\lambda_{N_i}, \phi_{N_i})$ such that*

$$\lim_{i\to\infty} \lambda_{N_i} = \lambda, \quad \phi_{N_i} \overset{w}{\to} \phi,$$

where $\lambda \in \mathbb{C}$ and $\phi \in \mathcal{F}$ are such that $\mathcal{K}\phi = \lambda\phi$. In particular if $\|\phi\| \neq 0$, then λ is an eigenvalue of $\mathcal{K}$ with eigenfunction ϕ.

Proof. First, observe that since $\mathcal{K}_N\phi_N = \lambda_N\phi_N$ with $\phi_N \in \mathcal{F}_N$, we also also have $P_N^\mu \mathcal{K} P_N^\mu \phi_N = \lambda_N\phi_N$. Hence $|\lambda_N| \leq \|P_N^\mu \mathcal{K} P_N^\mu\| \leq \|\mathcal{K}\| < \infty$ by Assumption 2 and the fact that $\|P_N^\mu\| \leq 1$. Therefore the sequence λ_N is bounded. Since ϕ_N is normalized and hence bounded, by weak sequential compactness of the unit ball of a Hilbert space there exists a subsequence $(\lambda_{N_i}, \phi_{N_i})$ such that $\lambda_{N_i} \to \lambda$ and $\phi_{N_i} \overset{w}{\to} \phi$.

It remains to prove that (λ, ϕ) is an eigenvalue-eigenfunction pair of $\mathcal{K}$. For ease of notation, set $\lambda_i = \lambda_{N_i}$ and $\phi_i = \phi_{N_i}$. Denote $\hat{\mathcal{K}}_i = \mathcal{K}_{N_i} P_{N_i}^\mu = P_{N_i}^\mu \mathcal{K} P_{N_i}^\mu$ and observe that $\hat{\mathcal{K}}_i\phi_i = \lambda_i\phi_i$ for all i. Then we have

$$\mathcal{K}\phi = \hat{\mathcal{K}}_i\left(\phi - \phi_i\right) + \left(\mathcal{K} - \hat{\mathcal{K}}_i\right)\phi + \hat{\mathcal{K}}_i\phi_i.$$

Taking the inner product with an arbitrary $f \in \mathcal{F}$ and using the fact that $\hat{\mathcal{K}}_i\phi_i = \lambda_i\phi_i$, we get

$$\langle \mathcal{K}\phi, f\rangle = \left\langle \hat{\mathcal{K}}_i\left(\phi - \phi_i\right), f\right\rangle + \left\langle \left(\mathcal{K} - \hat{\mathcal{K}}_i\right)\phi, f\right\rangle + \left\langle \lambda_i\phi_i, f\right\rangle.$$

Now, the second term on the right hand side $\left\langle \left(\mathcal{K} - \hat{\mathcal{K}}_i\right)\phi, f \right\rangle \to 0$ since $\hat{\mathcal{K}}_i$ converges strongly to $\mathcal{K}$ by Theorem 3 . The last term $\langle \lambda_i \phi_i, f\rangle \to \langle \lambda\phi, f\rangle$ since $\lambda_i \to \lambda$ and $\phi_i \xrightarrow{w} \phi$. It remains to show that the first term converges to zero. We have

$$\left\langle \hat{\mathcal{K}}_i \left(\phi - \phi_i\right), f \right\rangle = \left\langle P^{\mu}_{N_i} \mathcal{K} P^{\mu}_{N_i} \left(\phi - \phi_i\right), f \right\rangle = \left\langle \mathcal{K}\left(P^{\mu}_{N_i}\phi - \phi_i\right), P^{\mu}_{N_i} f \right\rangle,$$

where we used the fact that $P^{\mu}_{N_i}$ is self-adjoint and $\phi_i \in \mathcal{F}_{N_i}$ and hence $P^{\mu}_{N_i}\phi_i = \phi_i$. Denote $h_i := \mathcal{K}\left(P^{\mu}_{N_i}\phi - \phi_i\right)$. We will show that $h_i \xrightarrow{w} 0$. Indeed, denoting $\mathcal{K}^\star$ the adjoint of $\mathcal{K}$, we have

$$\left\langle \mathcal{K}\left(P^{\mu}_{N_i}\phi - \phi_i\right), f \right\rangle = \left\langle \left(P^{\mu}_{N_i}\phi - \phi + \phi - \phi_i\right), \mathcal{K}^\star f \right\rangle = \left\langle P^{\mu}_{N_i}\phi - \phi, \mathcal{K}^\star f \right\rangle + \langle \phi - \phi_i, \mathcal{K}^\star f\rangle \to 0,$$

since $P^{\mu}_{N_i}$ converges strongly to the identity (Lemma 2) and $\phi_i \xrightarrow{w} \phi$. Finally, we show that $\left\langle h_i, P^{\mu}_{N_i} f\right\rangle \to 0$. We have

$$\left\langle h_i, P^{\mu}_{N_i} f\right\rangle = \left\langle h_i, P^{\mu}_{N_i} f - f\right\rangle + \langle h_i, f\rangle .$$

The second term goes to zero since $h_i \xrightarrow{w} 0$. For the first term we have

$$\left\langle h_i, P^{\mu}_{N_i} f - f\right\rangle \leq \|h_i\| \left\|P^{\mu}_{N_i} f - f\right\| \to 0,$$

since $P^{\mu}_{N_i}$ converges strongly to the identity operator (Lemma 2) and h_i is bounded since $\mathcal{K}$ is bounded by Assumption 2, $\left\|P^{\mu}_{N_i}\right\| \leq 1$ and $\|\phi_i\| \leq 1$. Therefore we conclude that

$$\langle \mathcal{K}\phi, f\rangle = \lim_{i\to\infty} \langle \lambda_i \phi_i, f\rangle = \langle \lambda\phi, f\rangle$$

for all $f \in \mathcal{F}$. Therefore $\mathcal{K}\phi = \lambda\phi$. □

4.3.2 Simultaneous Convergence of the eDMD Operator

In this subsection we investigate an interesting analytical result when the number of sample points M and number of linearly basis functions N increase at the same rate, that is, $M = N \to \infty$. The space of observables we consider is $C(\mathcal{X})$ which we define as the space of functions f such that

$$\|f\|_{C(\mathcal{X})} = \sup_{x\in\mathcal{X}} |f(x)| < \infty.$$

which includes the continuous functions by the assumption below and the indication functions. Simply write $\mathcal{K}_N = \mathcal{K}_{N,N}$, with eigenvalues λ_N and eigenfunctions $\phi_N \in \mathcal{F}_N$. First we make some assumptions. The first assumption is to normalize the eigenfunctions and the rest are necessary for the convergence to be proven in this section.

Assumption 8. *Suppose that the following assumptions hold:*

1. *Suppose that $\mathcal{X}$ is compact and that all eigenfunctions of $\mathcal{K}_N$ are continuous.*
2. *The mapping T is a homeomorphism, i.e.,T is continuous and has continuous inverse T^{-1}*
3. *The samples x_i lie on the same trajectory.*

Thus, it is clear from point 1 of Assumption 8 that the eigenfunctions ϕ_N can be normalized. In this case since the matrices $\mathbf{Z}$ and $\mathbf{Z}'$ are square matrices, Assumption 6 then implies that the least squares problem can be solved exactly. That is, the equation

$$\mathbf{Z}' = A\mathbf{Z}$$

has a unique solution for $A \in \mathbb{R}^N$. Therefore, for any function $f \in \mathcal{F}_N$ this implies that the eDMD operator $\mathcal{K}_N$ is exact on these sample points. That is,

$$\left(\mathcal{K}f\right)(x_i) = \left(\mathcal{K}_{N,N}f\right)(x_i)$$

for all sample points x_i and $f \in \mathcal{F}_N$. for all $f \in \mathcal{F}_N$. Since the eigenfunctions are linear combinations of the basis functions, this relation holds for the eigenfunctions ϕ_N of $\mathcal{K}_{N,N}$, yielding

$$\left(\phi_N \circ T\right)(x_i) = \lambda_N \phi_N\left(x_i\right). \tag{4.37}$$

Using point 3 of Assumption 8, and by setting the additional point $x_{N+1} = T(x_N)$, we see that relation (4.37) can be written as

$$\frac{1}{N}\sum_{i=1}^{N} h(x_i)\phi_N(x_{i+1}) = \lambda_N \frac{1}{N}\sum_{i=1}^{N} h(x_i)\phi_N(x_i). \tag{4.38}$$

Furthermore, since a linear combination of Dirac masses is still a measure, by setting

$$\nu_N = \frac{1}{N}\sum_{i=1}^{N}\phi_N(x_i)\delta_{x_i},$$

the left hand side of (4.38) can be split up and expanded as

$$\begin{aligned}\frac{1}{N}\sum_{i=1}^{N} h(x_i)\phi_N(x_{i+1}) &= \frac{1}{N}\sum_{i=1}^{N} h(T^{-1}x_i)\phi_N(x_i) + \frac{1}{N}\left(h(x_N)\phi_N(x_{N+1}) - h(T^{-1}x_1)\phi_N(x_1)\right)\\ &= \int_{\mathcal{X}} h \circ T^{-1} d\nu_N + \frac{1}{N}\left(h(x_N)\phi_N(x_{N+1}) - h(T^{-1}x_1)\phi_N(x_1)\right).\end{aligned}$$

Therefore, substitution the left hand side in (4.38) gives

$$\int_{\mathcal{X}} h \circ T^{-1} d\nu_N + \frac{1}{N}\left(h(x_N)\phi_N(x_{N+1}) - h(T^{-1}x_1)\phi_N(x_1)\right) = \lambda_N \int_{\mathcal{X}} h d\nu_N. \tag{4.39}$$

Observe by points 1 and 2 of Assumption 8 that the $(h(x_N)\phi_N(x_{N+1}) - h(T^{-1}x_1)\phi_N(x_1))$ term is bounded and thus, converges to 0 over any subsequential limit. Indeed, by Prohorov's Theorem in Appendix A we can select a common subsequence $\nu_{N_i} \to \nu$ weakly and $\lambda_{N_i} \to \lambda$. Taking limits of both sides gives

$$\int_{\mathcal{X}} h \circ T^{-1} d\nu = \lambda \int_{\mathcal{X}} h d\nu \tag{4.40}$$

for all functions $h \in C(\mathcal{X})$. To interpret relation (4.40) we introduce invariant measures.

Definition 8. *A measure μ is said to be invariant on $\mathcal{X}$ with respect to a measureable map T if $\mu(T^{-1}(A)) = \mu(A)$ for all Borel sets $A \in \sigma(\mathcal{X})$).*

Remark 8. *An equivalent condition for a measure μ to be invariant on $\mathcal{X}$ with respect to a measurable map T is if $\int_{\mathcal{X}} f \circ T d\mu = \int_{\mathcal{X}} f d\mu$ for all countinuous bounded function f.*

Note that since the empirical measure $\hat{\mu}_N$ converges weakly to μ, it can be shown through some calculation under point 3 of Assumption 8 that the measure μ is invariant. Furthermore, it is a well known fact that when the measure μ is invariant that the dual of the Koopman operator (as an operator from $L_2(\mu) \to L_2(\mu)$) is given by

$$\mathcal{K}^* f = f \circ T^{-1}.$$

We show this here as well. This is because

$$\begin{aligned}
\langle \mathcal{K} f, g \rangle &= \int_{\mathcal{X}} (f \circ T) \bar{g} d\mu \\
&= \int_{\mathcal{X}} (f \circ T) \overline{(g \circ T^{-1} \circ T)} d\mu \\
&= \int_{\mathcal{X}} f \cdot \overline{(g \circ T^{-1})} d\mu \\
&= \langle f, \mathcal{K}^\star g \rangle ,
\end{aligned}$$

where the second and third lines follow from invariance of the measure and the skew symmetric property of the inner product. Therefore, (4.40) can be written as

$$\int_{\mathcal{X}} \mathcal{K}^\star h d\nu = \lambda \int_{\mathcal{X}} h d\nu. \tag{4.41}$$

Immediately this shows that the measure ν is an eigenmeasure of the Frobenius operator.

Definition 9. *The Perron-Frobenius operator $\mathcal{P} : M(\mathcal{X}) \to M(\mathcal{X})$, where $M(\mathcal{X})$ is the space of all complex-valued measures on $\mathcal{X}$, is defined for every $\eta \in M(\mathcal{X})$ and every Borel set A by*

$$(\mathcal{P}\eta)(A) = \eta\left(T^{-1}(A)\right).$$

Therefore, setting $h := g \circ T$ in gives

$$\int_{\mathcal{X}} g \circ T d\nu = \frac{1}{\lambda} \int_{\mathcal{X}} g d\nu.$$

Written out clearly, the measure ν is the eigenmeasure of the Perron-Frobenius operator with eiegnvalue $1/\lambda$. This is because we can write $g = 1_A$, which is the indicator function. Therefore, going this calculation would yield

$$\mathcal{P}\nu = \frac{1}{\lambda}\nu.$$

The Perron-Frobenius has been well known in the literature to be a dual to the Koopman operator, however it has received interest recently due to its connection with the Lyapunov density, a weakening of the Lyapunov function that verifies asymptotic stability almost everywhere with respect to some measure describing the distribution of the states.

Remark 9. *Note that by the measure theoretic version of the Riesz representation theorem, we can find representations for the certain functionals that look like eigenfunction relations. This is explained below, however it is still unclear how this could be useful.*

The idea will be to obtain generalized eigenfunctions of the L^2 adjoint of the Koopman operator as in the sense of [13]. Thus, we define the linear functionals $L_N : C(\mathcal{X}) \to \mathbb{C}$ by

$$L_N(h) = \int_{\mathcal{X}} h\phi_N d\hat{\mu}N,$$

and

$$\mathcal{K}L_N(h) = \int_{\mathcal{X}} h(\phi_N \circ T)\phi_N d\hat{\mu}N.$$

This means that relation (4.37) can be rewritten as

$$\mathcal{K}L_N = \lambda_N L_N. \tag{4.42}$$

Moreover, as we normalized the eigenfunctions so that $\|\phi_N\| = 1$ it is clear (by bounding by the sup pointwisely over the integral) that $\|L_N\| \leq 1$ and $\|\mathcal{K}L_N\| \leq 1$. As both functionals lie inside the unit ball, the Banach Anagolu theorem asserts that we for each sequence we can find a subsequence along which converge in the weak* topology to functionals $L \in C(\mathcal{X})$ and $\mathcal{K}L \in C(\mathcal{X})^*$ satisfying the relation

$$\mathcal{K}L = \lambda L,$$

where the scalar λ is a limit point of the sequence $\{\lambda_N\}$. Moreover, by the measure theoretic Riesz Representation theorem, the bounded linear functions L and $\mathcal{K}L$ can be represented by complex measures ν and $\mathcal{K}\nu$ such that

$$\mathcal{K}\nu = \lambda\nu. \tag{4.43}$$

Remark 10. *A note of caution that $\mathcal{K}L$ and $\mathcal{K}\nu$ are weak limits and its representation as a complex measure, rather than the action of the Koopman operator.*

4.3.3 Application of the Convergence of the eDMD Operator to Stability Analysis

The eigenfunctions of the Koopman operator can be used to form a Lyapunov function according to the formula

$$\mathcal{V}(x) = \left(\sum_{i=1}^{N} |\phi_{\lambda_i}(x)|^p \right)^{1/p},$$

with the integer $p \geq 1$. This is because these Lyapunov functions satisfy $\mathcal{V}\left(\varphi^t(x)\right) \leq \operatorname{Re}\left\{\lambda_1\right\} t\mathcal{V}(x)$ for $x \in X$, where λ_1 is the eigenvalue closest to the imaginary axis (which gives exponential stability). This result can be used to define a set

$$\Omega_\alpha = \{x \in X \mid V(x) < \alpha\},$$

which is forward invariant if $\Omega_\alpha \cap \partial X = \emptyset$, where ∂X is the boundary of X. In numerical simulations, the Lyapunov function can also be used to approximate the region of attraction. However, approximating the region of attraction using the eigenvectors of the eDMD operator will not yield good results, unless the number of sample points and the number of basis functions are large. A better numerical scheme which seeks to identify the Koopman eigenfunctions through solving a partial differential equation is described in [22].

We note that in recent years the Lyapunov density has become increasingly popular. While a Lyapunov function is a particular observable that decreases under the action of

$\mathcal{K}^t$, a Lyapunov density (or Lyapunov measure) decreases (almost everywhere) under the action of $\mathcal{P}^t$ [42]. The Lyapunov density was initially introduced in [32] as a function $C^1\left(X \backslash \{x^*\}\right)$ that satisfies $\nabla \cdot (F\rho) > 0$, a property which precisely corresponds to the action of the Perron-Frobenius infinitesimal generator $L_P\rho < 0$. In future work we will identify the connection between the action of the Perro-Frobenius operator and the action of its infinitesimal generator and use the results of Section 4.3.2 to derive guarantees on the existence of Lyapunov densities for this method.

4.4 Infinite Matrix Representation of the Koopman Operator

In this section we show that the Koopman operator admits a matrix representation in a suitably chosen basis. Let the state space X be a compact forward invariant set. Correspondingly on this set, consider $L^2(X)$ that is $f \in L^2(X)$ if it satisfies

$$\int_X f^2(x)dx < \infty \tag{4.44}$$

and it can be shown that $L^2(X)$ is a Hilbert space with norm generated by the following inner product

$$\langle f, g \rangle := \int_X f(x)\overline{g(x)}dx. \tag{4.45}$$

The motivation of this representation comes from the following classical result in the theory of Hilbert spaces.

Theorem 15. *Let H be a Hilbert space. The following are equivalent:*

- *H is separable.*
- *H has a countable orthonormal basis.*
- *Every orthonormal basis for H is countable.*

In addition to considering the special case of square integrable functions, we require assumptions on the flow operator. This is formulated in the following theorem

Theorem 16. *Let X be a forward invariant and compact set. Fix $t > 0$ and suppose that the flow operator Φ^t is continuously differentiable. Then the Koopman operator $\mathcal{K}^t$ is a bounded linear operator. Moreover, if $\{\phi_1, \ldots, \phi_n, \ldots\}$ represents the Schauder basis formed from Hermitian polynomials then we have the following matrix representation*

$$\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\langle \mathcal{K}^t\phi_k, \phi_j\rangle\langle f, e_k\rangle = \mathcal{K}^t f \tag{4.46}$$

for all observables $f \in L^2(X)$. The coefficients $\langle \mathcal{K}^t\phi_k, \phi_j\rangle$ are referred to as the matrix coefficients of $\mathcal{K}^t$.

Proof. Note by the forward invariance of X we have that $\Phi^t(x) \in X$ for all $x \in X$ so the following mapping is well defined

$$\mathcal{K}^t : L^2(X) \to L^2(X). \tag{4.47}$$

Before proving that $\mathcal{K}^t$ is a bounded linear operator note that since Φ^t is continuously differentiable, and X is a compact set, by the Weierstrass theorem it follows that $|\det J_{(\Phi^t)^{-1}}| \leq M$ for some $M > 0$. The change of variables theorem gives that

$$\begin{aligned}
\|\mathcal{K}^t f\|_2^2 &= \int_X |f \circ \Phi^t|^2 dx \\
&= \int_{\Phi^{-1}(X)} |\det J_{(\Phi^t)^{-1}}(u)||f \circ \Phi^t \circ \Phi^{t^{-1}}(u)|^2 du \\
&= \int_{\Phi^{-1}(X)} |\det J_{(\Phi^t)^{-1}}(u)||f|^2 du \\
&\leq M\|f\|_2^2
\end{aligned}$$

and therefore the operator $\mathcal{K}^t$ is a bounded linear operator. It can be shown that the family of Hermitian polynomials $\{\phi\}_n$ is an orthonormal basis and therefore by the previous theorem, it must form a countable basis of $L^2(X)$. Now, let P_n denote the projection operator onto the subspace spanned by $\{\phi_1, \ldots, \phi_n\}$. Since $P_n \to Id$ in the strong sense this implies that $P_n\mathcal{K}^tP_mf \to P_n\mathcal{K}^tf$ in the strong sense. Thus, the coordinates in this basis is given by

$$\lim_{m\to\infty}\langle \mathcal{K}^t(\sum_{k=1}^{m}\langle f, \phi_k\rangle\phi_k), \phi_j\rangle = \sum_{k=1}^{\infty}\langle \mathcal{K}^t\phi_k, \phi_j\rangle\langle f, \phi_k\rangle.$$

Note that convergence holds here since the operator $\mathcal{K}^t$ is a bounded linear operator. Finally, taking $P_n\mathcal{K}^t f \to \mathcal{K}^t f$ implies that

$$\sum_{j=1}^{\infty}\sum_{k=1}^{\infty}\langle \mathcal{K}^t\phi_k, \phi_j\rangle\langle f, \phi_j\rangle\phi_j = \mathcal{K}^t f.$$

□

Similar to the methodology of the extended DMD, we can sample data points to numerically approximate the inner product $\langle \mathcal{K}^t\phi_k, \phi_j\rangle$. The advantage of this method is the fast evaluation of new observable functions. The application of this representation will be investigated in future work in the context of Lyapunov barrier functions.

Chapter 5

Estimating the Region of Attraction

Recall that the ROA of an asymptotically stable equilibrium point as defined in Definition 2 is the set of all states that tend to an asymptotically stable equilibrium point. For practical systems, the region of attraction is typically a proper subset of the entire state space and therefore estimating a sufficiently large ROA allows for practical systems to operate at a sufficiently stable operating point. Therefore, developing efficient numerical schemes for estimating the ROA has been studied extensively [1, 6, 21, 26]. Analytically, the celebrated Hartman-Grobman theorem asserts that the Jacobian can be used to test for stability of some neighbourhood of the equilibrium point. However, in this chapter we review a general methodology for efficiently computing the ROA as described in [45].

In [45] the authors discuss estimating the exact ROA. The exact ROA is not a closed form description of that set, but rather the entire ROA is obtained indirectly via a set that is convenient for numerical calculations (such as the largest level set of a Lyapunov function that lies in a valid region). The traditional method for estimating the ROA is called Zubov's method [10, 43, 14]. Zubov's method represents the ROA via the optimal Lyapunov function which is the solution of a first-order partial differential equation (PDE).

Each Lyapunov function $V(x)$ associated with an equilibrium point x_s yields an estimate of the ROA contained in the exact ROA. However, there does exist a Lyapunov function whose associated ROA exactly coincides with the exact ROA. Zubov's theorem provides a way to determine this Lyapunov funciton through solving a PDE.

Theorem 17. *(Zubov's Theorem). Given an autonomous dynamical system $\dot{x} = f(x)$ with $f : \mathcal{X} \subset \mathbb{R}^n \to \mathbb{R}$ and an equilibrium point $x_s \in \mathcal{X}^o$, a set $\mathcal{R} \subset \mathcal{X}$ with $x_s \in \mathcal{R}^\circ$ is the ROA of x_s iff $\exists u, h$ such that:*

1. *$V(0)=h(0)=0, 0<V(x)<1$ for $x \in \mathcal{R} \backslash\{0\}$, and $h>0$ for $x \in \mathcal{X} \backslash\{0\}$.*
2. *$\forall \gamma_2>0, \exists \gamma_1>0$ and $\alpha_1>0$ such that $u(x)>\gamma_1$ and $h(x)>\alpha_1$ if $\|x\|>\gamma_2$.*
3. *$V\left(x_n\right) \rightarrow 1$ for $x_n \rightarrow \partial \mathcal{R}$ (or $\left\|x_n\right\| \rightarrow \infty$).*
4. *$\nabla V(x) \cdot f(x)=-h(x)(1-u(x)) \sqrt{1+\|f(x)\|^2}$.*

Note that points 1 and 4 concern positive and negative definiteness respectively of $V(x)$ in satisfying the Lyapunov conditions. The central point is point 3 and it says that the solution to Zubov's PDE yields a Lyapunov function $V(x): \mathcal{X} \subset \mathbb{R}^n \rightarrow \mathbb{R}$ whose associated ROA is the complete ROA). However, this result is rather limited due to Zubov's PDE does not having an analytical solution in general. Nonetheless, several different numerical approaches have been developed to compute an approximate solution to Zubov's PDE, and exact solutions have been developed for specific applications [33].

The main difficulty with applying Zubov's method is the auxiliary function h has to be provided prior to solving for the Lyapunov function $V(x)$. Fortunately there are other ways to determine an implicit, level set based representation of the complete ROA associated with an equilibrium point. One such method described in the following section depends on finding the viscosity solution to a particular Hamilton-Jacobi PDE.

5.1 Theoretical Foundations

In this section we denote the solution to the problem (3.1) as $x(t; x_0, t_0)$ where x_0 and t_0 are to be understood as the initial states and times respectively. We now present an alternative way to determine the exact ROA, but to do this we need to introduce two definitions.

Definition 10. *For system (3.1), the forward reachable set over the time interval $[0, t]$ is defined as*

$$F_f(I,[0, t]):=\left\{x \in \mathbb{R}^n \mid \exists x_0 \in I, \exists s \in[0, t], x\left(s ; x_0, 0\right)=x\right\} .$$

Intuitively, the forward reachable set can be thought of as the states for the trajectory can reach over a fixed time interval t. Note that since this is an autonomous system, the specific end points of the interval have no effect on the forward reachable set. Similarly, the backward reachable set is the set containing the sets for which the trajectories can reach a target set within a given time interval. The backward reachable set is generally defined by a terminal-value problem

$$B_f(K,[-t, 0]):=\left\{x_0 \in \mathbb{R}^n \mid \exists x \in K, \exists s \in[-t, 0], x\left(s ; x_0,-t\right)=x\right\},$$

where $K \subset \mathbb{R}^n$ is the target set. Similarly, the specific endpoints have no effect on the backward reachable set for autonomous systems. In this section, the ROA is represented implicitly as the zero sub-level set of a level set function. We denote the zero sub-level set as

$$S_0(f) := \{x \in \mathbb{R}^n : f(x) \leq 0\},$$

and the zero level set is denoted as

$$S_0(f) := \{x \in \mathbb{R}^n : f(x) = 0\}.$$

The connection between the ROA of an asymptotically stable equilibrium point x_s and the backward reachable set of subset Ω of the ROA is clear. Since all states in the ROA must tend towards x_s as time tends to infinity, the time horizon for the backward reachable set with Ω as a target set must be infinite. This is clearly stated in the following lemma for which we offer a simpler proof.

Lemma 5. *Let x_s be an asymptotically stable equilibrium point of system (3.1) and $A_f(x_s)$ denote the ROA of x_s. Then, let $\Omega \subset \mathbb{R}^n$ be a closed region such that*

$$\begin{array}{ll} x_s & \in \Omega^\circ, \\ \Omega & \subset A_f\left(x_s\right), \end{array}$$

where Ω° denotes the interior of Ω. Then

$$A_f\left(x_s\right) = B_f(\Omega, (-\infty, 0]),$$

where $B_f(\Omega, (-\infty, 0])$ is defined analogously as in Definition , with the exception of the interval $(-\infty, 0]$ in the place of the compact interval $[-t, 0]$.

Proof. Fix $x_0 \in A_f\left(x_s\right)$. Since Ω° is open there exists $\epsilon > 0$ such that $B_\epsilon(x_s) \in \Omega^\circ$. By definition of the ROA, there exists a time $T > 0$ such that $x(t; x_0, 0) \in B_\epsilon(x_s)$ for all $t \geq T$. Letting $x_T = x(T; x_0, 0)$ we see that $x_T \in \Omega$ and $x_T \in B_f(\Omega, (-\infty, 0])$. This proves that $A_f\left(x_s\right) \subset B_f(\Omega, (-\infty, 0])$.

Conversely, fix $x_0 \in B_f(\Omega, (-\infty, 0])$. This means that there exists $T > 0$ and $x \in \Omega$ such that $x = x(T; x_0, 0)$. But since $x(t; x, 0) \to x_s$ as $t \to \infty$ we have that $x_0 \in A_f\left(x_s\right)$ which proves that $B_f(\Omega, (-\infty, 0]) \subset A_f\left(x_s\right)$. $\square$

Lemma 5.1 states that the ROA can be obtained by the evolution of the backward reachable set containing x_s and contained in the ROA. This result leads to the development of a methodology which has the advantage of not needing to choose an auxiliary function as in Zubov's method. The proposed method employs a time dependent PDE leading to an iterative algorithm. There are multiple methods to numerically approximate the backward reachable set based on a stationary Hamilton-Jacobi equation [41, 38]. In particular, the backward reachable set can be computed as a limit of sub-level set of the viscosity solution of the Hamilton-Jacobi equation. A special case of Theorem 2 in [24] states this below.

Theorem 18. *Let $\phi(x,t) : \mathbb{R}^n \times (-\infty, 0] \to \mathbb{R}$ be the viscosity solution of the terminal value Hamilton-Jacobi equation*

$$\begin{aligned} \tfrac{\partial \phi}{\partial t} + \min\left[0, \left(\tfrac{\partial \phi}{\partial x}\right)^T f(x)\right] &= 0, \\ \phi(x,0) &= \phi_0(x), \end{aligned} \tag{5.1}$$

where $\phi_0(x)$ is bounded and Lipschitz continuous, and

$$\begin{aligned} x_s &\in S_0\left(\phi_0\right), \\ S_0\left(\phi_0\right) &\subset A_f\left(x_s\right), \end{aligned} \tag{5.2}$$

Then, the ROA can be described by ϕ

$$B_f(\{x \in \mathbb{R}^n : \phi_0(x) \leq 0\}, (-\infty, 0]) = \{x \in \mathbb{R}^n \mid \phi(x, -\infty) < 0\}, \tag{5.3}$$

where $\phi(x,-\infty) = \lim_{\tau \to \infty} \phi(x, -\tau)$.

The previous two results then immediately imply the following corollary.

Corollary 1. *Given the same assumptions as in Theorem 18, the ROA can be described by*

$$A_f\left(x_s\right) = \{x \in \mathbb{R}^n \mid \phi(x, -\infty) < 0\}. \tag{5.4}$$

Corollary 1 encodes the computation of the ROA as a terminal value Hamilton–Jacobi equation. Since the viscosity solution $\phi(x,t)$ changes continuously over time, its evolution backwards in time implies that the sub-level set $S_0(\phi(x,t))$ is getting closer to the exact ROA.

Remark 11. *Note that the sub-level sets $S_0(\phi(x,t))$ increase in time and are always subsets of the exact ROA. Indeed, since $\frac{\partial \phi}{\partial t} \geq 0$, for $0 < T < \infty, \phi(x,-T) \geq \phi(x,-\infty)$, which means $S_0(\phi(x,-T)) \subset \{x \in \mathbb{R}^n \mid \phi(x,-\infty) < 0\}$, therefore $S_0(\phi(x,-T)) \subset A_f\left(x_s\right)$.*

The remark implies that for sufficiently large time T, $S_0(\phi(x,-T))$ can be used as a conservative estimate of the exact ROA.

5.2 Methodology

As proposed in [45], we state their evolutionary methodology to determine the ROA based on the theoretical foundations of the previous section. The proposed methodology tracks the evolution of the viscosity solution of the Hamilton-Jacobi equation on a fixed Cartesian grid in any number of dimensions. This is an iterative method that yields a larger estimate of the ROA than the previous time point. The following is an outline of the methodology for determining the ROA:

Algorithm

1. Create the computational grid. The state space is discretized into a fixed Cartesian grid, on which the estimates of the ROA evolve.

2. Compute the vector field on the grid. Compute the vector field of the system $\dot{x} = f(x)$ on the grid created in Step 1. The vector field is used to construct the numerical Hamiltonian.

3. Create the initial condition. The zero sub-level set of the initial condition is an initial estimate of the ROA and the zero level-set of initial conditions is an initial estimate of the boundary. A quadratic Lyapunov function based on the linearized system can be used to generate the initial condition. Additionally, a signed distance function $\phi_0(x) = \|x - x_s\|_2 - c$ can also be employed as the initial condition, where the constant $c \in \mathbb{R}^+$ is the radius of the initial spherical domain.

4. Compute the viscosity solution of the underlying Hamilton-Jacobi equation. Integrate the underlying Hamilton-Jacobi equation backwards in time. Set a computational period, say ΔT, and a limit on the total evolution time, say T_z. A new level set function $\phi(x, t - \Delta T)$ and a new estimate $S_0(\phi(x, t - \Delta T))$ can be obtained. Then, let $\phi(x, t - \Delta T)$ be the initial condition and integrate the underlying equation until the level set $\phi(x, t)$ remains unchanged or the total evolution time reaches T_z.

In this method, there is no optimal method for choosing the computational grid in Step 1. However, it is typically taken as a uniform grid over the state space and the increments can be decreased by trial and error to achieve a higher degree of accuracy. In Step 3, since $\frac{\partial \phi}{\partial t} \geq 0$ we need to provide an initial condition such that its zero sub-level set contains the asymptotically stable equilibrium point and is contained in the ROA. A conservative estimate of the ROA can be determined by linearizing the system about x_s. In particular, we want to compute a quadratic Lyapunov function and use its sub-level set to generate an initial underestimate of the ROA (typically this produces a conservative underestimation). Letting A denote the Jacobian of the system (3.1) about x_s, we solve the Lyapunov equation for P

$$PA + A^{\mathrm{T}}P = -Q. \tag{5.5}$$

Here Q is a symmetric positive definite matrix which is usually taken as the identity matrix. Since x_s is an asymptotically stable equilibrium point, the matrix A is Hurwitz (i.e. possessing eigenvalues with negative real part) so the Lyapunov equation has a unique symmetric solution ([17]). Thus, the level set of the quadratic Lyapunov function $V(x) = x^T Px$ can be used to estimate $A_f(x_s)$. Consider the sublevel set $\Omega_c = \left\{x^T Px - c \leq 0\right\}$ for some $c > 0$. If $\dot{V}(x) = 2f(x)^{\mathrm{T}}Px$ is negative definite in $\{\Omega_c - x_s\}$, then $\Omega_c \subset A_f\left(x_s\right)$. Hence $\phi_0(x) = x^{\mathrm{T}}Px - c$ can be used as an initial condition and a suitable c can be determined by computing $2f(x)^{\mathrm{T}}Px$ and which fits the largest level set in the valid region.

The key part of this method is the choice of the numerical method that computes the viscosity solution of the Hamilton-Jacobi method. Several methods have been proposed for this, but the level set method [25] is extensively used due to its efficient scheme. In the level set method, the state space is discretized into a grid and the solution of the Hamilton-Jacobi equation is represented implicitly through a level set function $\phi(x, t)$, hence the name of this method. Due to the inherent sensitivy of this method to spacial accuracy, the high-order accurate method is employed to estimate spatial derivatives $\partial\phi/\partial x$ up to high accuracy. Therefore, Equation (5.1) can be rewritten as

$$\frac{\partial \phi}{\partial t} = -\min\left[0, \left(\frac{\partial \phi}{\partial x}\right)^T f(x)\right]. \tag{5.6}$$

Thus, once the Hamiltonian is obtained, the value of $\phi(x, t)$ can be obtained on each grid node by time integration via explicit Runge-Kutta schemes.

5.3 Numerical Implementation

Here we present some examples employing the method outlined in Section 5.2. Moreover, we compare the results of the numerical experiments of this section to those of Section 3.3.

5.3.1 Van Der Pol Oscillator

We recall the equations of motion for the Van Der Pol Oscillator are

$$\begin{aligned}\dot{x}_1 &= -x_2 \\ \dot{x}_2 &= x_1 + \left(x_1^2 - 1\right) x_2.\end{aligned} \tag{5.7}$$

We choose the Vander Pol Oscillator as it is a simple and serves as a good comparison as the dynamics does not involve an input u. Firstly, we check the consistency of the algorithm described in Section 5.2 by first comparing the phase plot.

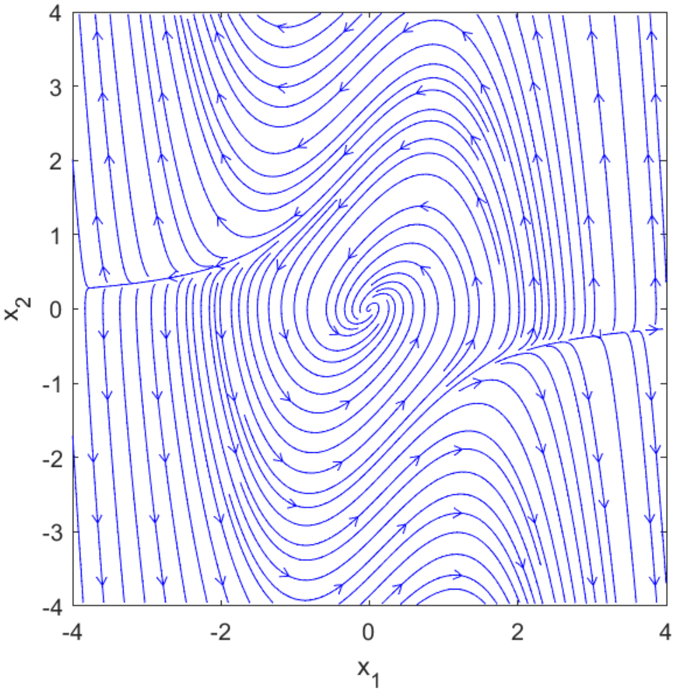


Figure 5.1: Phase space plot of the Val Der Pol oscillator without controller. From this image we can form an idea for the actual ROA.

Now that the phase plot is consistent, the computational grid is a rectangular Euclidean mesh bounded by $x_1 \in [-4, 4]$ and $x_2 \in [-4, 4]$, the size of grid cell is set to 0.01. Since the equations of motion are simple, we simply choose the initial condition as the function ϕ_0 that has the circle of radius 0.5 centered at the origin as its level curve. Here in Figure 5.2, the estimates of the ROA and boundary of different evolutionary times are plotted. In that figure, ten curves are plotted with each curve differing by an evolution time of 1 second. The terminal evolutionary time corresponds to the largest curve at time 10 seconds. Figure 5.3 plots the estimated boundary obtained by the proposed methodology after 10 seconds of evolution onto the phase portrait.

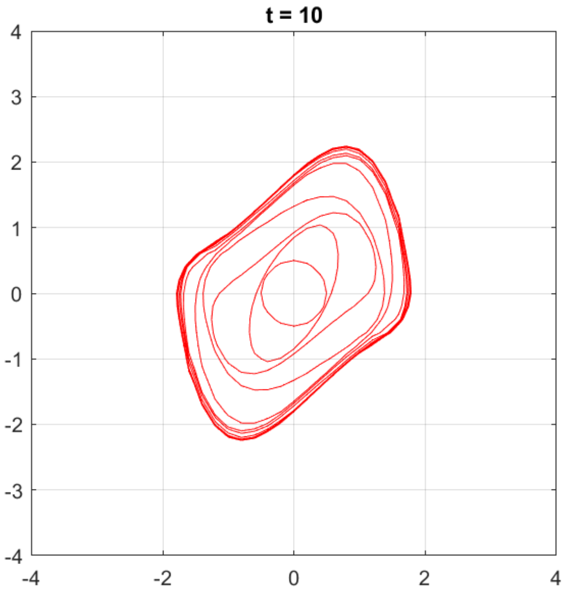


Figure 5.2: Estimates of the ROA and boundary of different evolutionary time

Note that compared to Section 3.3, the estimates of the ROA are much larger and closely match the actual ROA of the Van Der Pol Oscillator. However, the down side of this method is that the level set produces a set that does not appear to be forward invariant with respect to the phase portrait. Moreover, unlike the learning methodology this process requires a well modeled differential equation and so, in the presence of data, the dynamics first have to be learned and there is no way to obtain formal guarantees that the simulated ROA is close (in some kind of set metric, with a possible candidate being the Hausdorff distance) to the actual ROA.

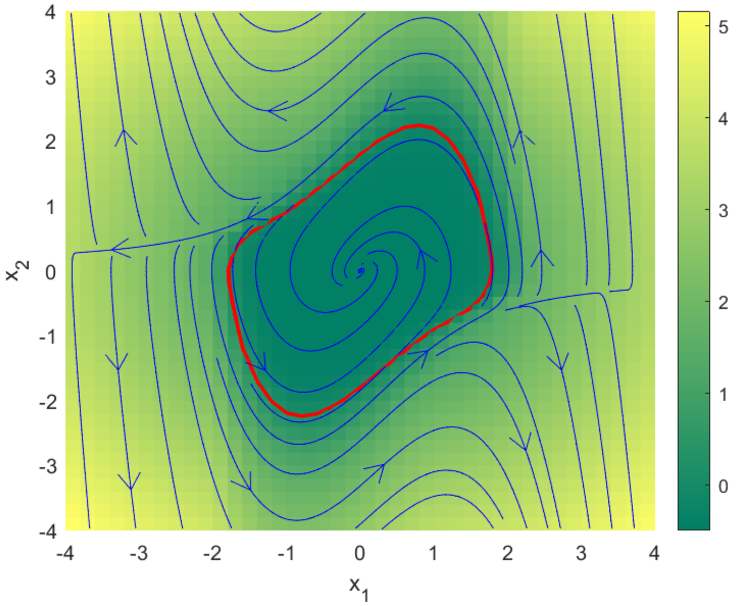


Figure 5.3: The level set function $\phi(x, 10)$ and the phase portrait of the system.

5.3.2 Chiang and Thorp Differential Equation

This is an example studied in Chiang and Thorp [8]. Here $x_s = (0.02801, 0.06403)$ is the equilibrium point whose ROA is to be determined. The equations of motion are given by

$$\begin{aligned} \dot{x}_1 &= -\sin x_1 - 0.5 \sin(x_1 - x_2) + 0.01, \\ \dot{x}_2 &= -0.5 \sin x_2 - 0.5 \sin(x_2 - x_1) + 0.05. \end{aligned} \tag{5.8}$$

Applying the proposed methodology to this system, the computational grid is a rectangular Euclidean mesh bounded by $x_1 \in [-6, 6]$ and $x_2 \in [-6, 6]$, the size of grid cell is set to 0.01. In this example, the initial condition is computed by solving the Lyapunov equation with the Jacobian matrix A and symmetric positive definite matrix Q. Then

$$A \approx \begin{bmatrix} -1.5 & 0.5 \\ 0.5 & -1 \end{bmatrix}, \quad Q = \begin{bmatrix} 3 & -2 \\ -2 & 2 \end{bmatrix}, \quad P \approx \begin{bmatrix} 0.84 & -0.48 \\ -0.48 & 0.76 \end{bmatrix}.$$

The following function provides an initial condition

$$\phi(x, 0) = x^{\mathrm{T}} P x - 1.$$

A sequence of ROA estimates of different evolutionary times is plotted in Figure 5.4 (a). Curve D (in black) is the exact boundary. Curve A is the zero-level set of the initial condition. Curve B is the estimated boundary obtained by the proposed methodology after 2 seconds of evolution and curve C is the estimated boundary after 10 seconds of evolution. We note that the boundary in curve C is almost indistinguishable from the exact boundary. Figure 5.4 (b) is the level set function $\phi(x, 10)$ and the phase portrait. Indeed, the sequence of estimated ROA is an increasing sequence which is contained in and converges to the exact ROA. Here we see that the level set appears to be forward invariant with respect to the phase portrait. Therefore, invariance seems to be problem dependent and it would be interesting to develop conditions that guarantee invariance of the level set computed by this methodology.

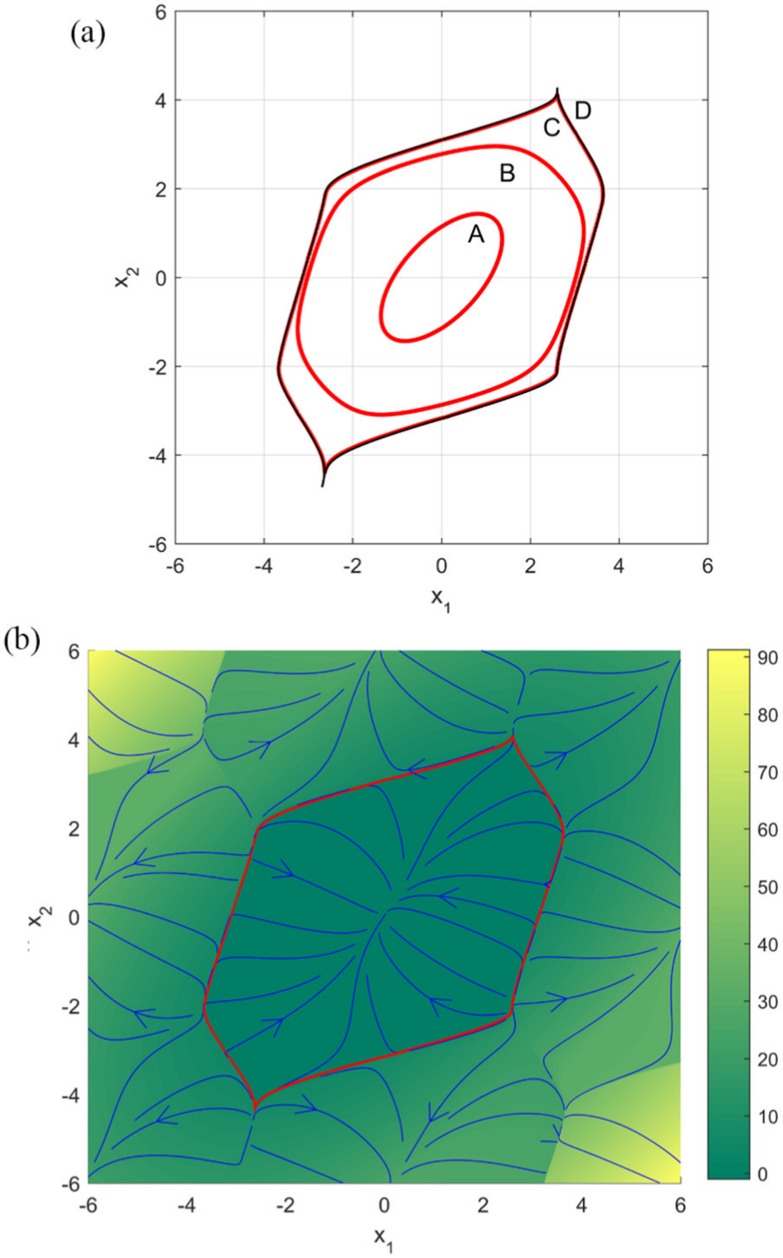


Figure 5.4: (a) Estimates of the ROA and boundary of different evolutionary time. (b) The level set of the curve $\phi(x, 10)$ and the phase portrait of the system.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

In Chapter 3, we developed a methodology for performing system identification given measurements of the vector field and learned a valid Lyapunov function based on the learned dynamics. The main innovation of this method is performing verification with SMT as the main tool and developing a method using the Lipschitz constants for verifying the validity of the learned Lyapunov function for the actual dynamics. The main difficulty of using this methodology is due to the inability of the SMT solver to verify the Lyapunov conditions in some neighborhood of the origin. Therefore, we introduced assumptions under which some technical results can be proven so that the learned Lyapunov function achieves practical stability about the origin. Another potential downside of this method is the inability to scale to a high dimensional setting. This method suffers from the curse of dimensionality for the same reason as many popular statistical learning methods such as least squares. Simply put, in higher dimensions more data points are required to achieve the same separation and this leads to massive training times.

To counter this problem of the neural network method, in Chapter 4, we introduce the eDMD algorithm which is numerically very efficient as it is recasting least squares in the context of finding a high dimensional linear representation of the dynamics. The eigenfunctions of the Koopman operator can be used as Lyapunov functions for the system (assuming the real part of the eigenvalues are negative). We subsequently, used the results regarding the convergence of the eDMD operator to the Koopman operator to develop guarantees for these Lyapunov functions. Finally, in Chapter 5, we introduced a method for determining the ROA of a dynamical system based on solving the Yuan-Li PDE. This method works well, however due to approximation errors that occur when solving a PDE numerically, there is no guarantee that the obtained region will be forward invariant.

6.2 Future Work

The research presented in all chapters can still be improved upon. We suggest the following as questions that could potentially lead to interesting new results and research:

1. The data set used to train the unknown dynamics and learn the Lyapunov function is generated from the trajectories of solutions to ordinary differential equations, but in practice the actual measurements of the states are typically noisy, and sometimes it is difficult to have direct access to the states measurements and obtain a significant number of data points. For the learning methodology to become useful, we need to understand how to learn the unknown dynamics and a robust Lyapunov function with different values of β in (3.9) to guarantee stability with noisy measurements. Furthermore, the implementation of this algorithm on real dynamical systems will be investigated as well afterwards.

2. To complement the analysis in Section 4.3.1 and Section 4.3.2, we should focus on non-asymptotic analysis of optimal basis functions. That is, for any $n \in \mathbb{N}$, how should the basis functions $f \in \mathcal{F}_N$ be chosen to minimize some metric. One possible methodology would be to learn the basis functions with neural networks with N functions outputted in parallel and trained to minimize $\left\|\mathcal{K}_N - \mathcal{K}_{|\mathcal{F}_N}\right\|$ as the loss function. In this problem, the loss function should also contain terms such that the space $\mathcal{F}_N$ should contain observables that provide 'rich' measurements of the state. Providing a formal definition of this and encoding this into the loss function is a challenging task. Therefore, it be interesting to compare a successful learning algorithm with the existing attempts at choosing the basis functions.

3. More thought should be given to the results of Section 4.3.1 and Section 4.3.2 to formulate formal guarantees on the existence of Lyapunov functions and Lyapunov densities respectively. To the best of my knowledge, no rigorous guarantees on the existence of a Lyapunov function or a Lyapunov density computed by the eDMD algorithm exists in the literature. Formulating such a result will offer insight into the types of problems that can possibly yield Lyapunov functions via the eDMD algorithm.

4. Learning based methods, especially those related to physics informed neural networks (PINNs) typically used to solve PDEs should be used to learn the auxiliary function while simultaneously solving Zubov's PDE to obtain the Lyapunov function. It is possible that by tweaking the loss functions and structure of the neural network that the same methodology used for our NeurIPS submission can be adapted to this problem.

References

[1] Marian Anghel, Federico Milano, and Antonis Papachristodoulou. Algorithmic construction of lyapunov functions for power system stability analysis. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 60(9):2533–2546, 2013.

[2] Patrick Billingsley. *Convergence of probability measures*. Wiley Series in Probability and Statistics: Probability and Statistics. John Wiley & Sons Inc., New York, second edition, 1999. A Wiley-Interscience Publication.

[3] Steven L. Brunton, Marko Budišić, Eurika Kaiser, and J. Nathan Kutz. Modern koopman theory for dynamical systems. *SIAM Review*, 64(2):229–340, 2022.

[4] Steven L. Brunton and Bernd R. Noack. Closed-Loop Turbulence Control: Progress and Challenges. *Applied Mechanics Reviews*, 67(5), 08 2015. 050801.

[5] Ricardo Carona, A Pedro Aguiar, and Jos Gaspar. Control of unicycle type robots tracking, path following and point stabilization. In *International Proceeding of IV Electronics and Telecommunications*. Citeseer, November 2008.

[6] Abhijit Chakraborty, Peter Seiler, and Gary J. Balas. Nonlinear region of attraction analysis for flight control verification and validation. *Control Engineering Practice*, 19(4):335–345, 2011.

[7] Ya-Chien Chang, Nima Roohi, and Sicun Gao. Neural lyapunov control. *Advances in neural information processing systems*, 32, 2019.

[8] H.-D. Chiang and J.S. Thorp. Stability regions of nonlinear dynamical systems: a constructive methodology. *IEEE Transactions on Automatic Control*, 34(12):1229–1241, 1989.

[9] Alessandro Chiuso and Gianluigi Pillonetto. System identification: A machine learning perspective. *Annual Review of Control, Robotics, and Autonomous Systems*, 2:281–304, 2019.

[10] Rodney D. Driver. Methods of a. m. lyapunov and their application (v. i. zubov). *SIAM Rev.*, 7(4):570–571, oct 1965.

[11] A. F. Filippov. *Differential equations with discontinuous righthand sides*, volume 18 of *Mathematics and its Applications (Soviet Series)*. Kluwer Academic Publishers Group, Dordrecht, 1988. Translated from the Russian.

[12] G.B. Folland. *Real Analysis: Modern Techniques and Their Applications*. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts. Wiley, 2013.

[13] I M Gelfand, Georgi E Shilov, M I Graev, N Y Vilenkin, and I I Pyatetskii-Shapiro. *Generalized functions*. AMS Chelsea Publishing. Academic Press, New York, NY, 1964. Trans. from the Russian, Moscow, 1958.

[14] Wolfgang Hahn. *Stability of Motion*. Grundlehren der mathematischen Wissenschaften. Springer Berlin, Heidelberg, 1967.

[15] Yu Jiang and Zhong-Ping Jiang. *Robust adaptive dynamic programming*. John Wiley & Sons, 2017.

[16] Dirk M. Luchtenburg Steven L. Brunton J. Nathan Kutz Jonathan H. Tu, Clarence W. Rowley. On dynamic mode decomposition: Theory and applications. *Journal of Computational Dynamics*, 1:391–421, 2014.

[17] H.K. Khalil. *Nonlinear Control*. Always Learning. Pearson, 2015.

[18] Milan Korda and Igor Mezić. On convergence of extended dynamic mode decomposition to the koopman operator. *Journal of Nonlinear Science*, 28(2):687–710, nov 2017.

[19] Xin Li. Simultaneous approximations of multivariate functions and their derivatives by neural networks with one hidden layer. *Neurocomputing*, 12(4):327–343, 1996.

[20] Yuandan Lin, Eduardo D Sontag, and Yuan Wang. A smooth converse lyapunov theorem for robust stability. *SIAM Journal on Control and Optimization*, 34(1):124–160, 1996.

[21] M. L. Matthews and C. M. Williams. Region of attraction estimation of biological continuous boolean models. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 1700–1705, 2012.

[22] Alexandre Mauroy and Igor Mezić. Global stability analysis using the eigenfunctions of the koopman operator. *IEEE Transactions on Automatic Control*, 61(11):3356–3369, 2016.

[23] Igor Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41:309–325, 2005.

[24] I.M. Mitchell, A.M. Bayen, and C.J. Tomlin. A time-dependent hamilton-jacobi formulation of reachable sets for continuous dynamic games. *IEEE Transactions on Automatic Control*, 50(7):947–957, 2005.

[25] Stanley J. Osher and Ronald Fedkiw. *Level set methods and dynamic implicit surfaces.*, volume 153 of *Applied mathematical sciences*. Springer, 2003.

[26] Rohit Pandita, Abhijit Chakraborty, Peter Seiler, and Gary Balas. Reachability and region of attraction analysis applied to gtm dynamic flight envelope assessment. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Guidance, Navigation, and Control Conference and Exhibit. American Institute of Aeronautics and Astronautics Inc., 2009.

[27] P.A. Parrilo, California Institute of Technology. Division of Engineering, and Applied Science. *Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization*. CIT theses. California Institute of Technology, 2000.

[28] Xue Bin Peng, Erwin Coumans, Tingnan Zhang, Tsang-Wei Edward Lee, Jie Tan, and Sergey Levine. Learning agile robotic locomotion skills by imitating animals. In *Robotics: Science and Systems*, 07 2020.

[29] Gianluigi Pillonetto, Francesco Dinuzzo, Tianshi Chen, Giuseppe De Nicolao, and Lennart Ljung. Kernel methods in system identification, machine learning and function estimation: A survey. *Automatica*, 50(3):657–682, 2014.

[30] Allan Pinkus. Approximation theory of the mlp model in neural networks. *Acta Numerica*, 8:143–195, 1999.

[31] Joshua L. Proctor and Philip A. Eckhoff. Discovering dynamic patterns from infectious disease data using dynamic mode decomposition. *Int Health.s*, 7(2):139–145, 2015.

[32] Anders Rantzer. A dual to lyapunov's stability theorem. *Systems Control Letters*, 42(3):161–168, 2001.

[33] Mohaamad Rezaiee Pajand and Behrang Moghaddasie. Estimating the region of attraction via collocation for autonomous nonlinear systems. *Structural Engineering and Mechanics*, 41(2):263–284, February 2012.

[34] H.L. Royden and P. Fitzpatrick. *Real Analysis*. Prentice Hall, 2010.

[35] W. Rudin. *Functional Analysis*. International series in pure and applied mathematics. McGraw-Hill, 1991.

[36] PETER J. SCHMID. Dynamic mode decomposition of numerical and experimental data. *Journal of Fluid Mechanics*, 656:5–28, 2010.

[37] J.T. Schwartz and H. Karcher. *Nonlinear Functional Analysis*. Notes on mathematics and its applications. Gordon and Breach, 1969.

[38] J. A. Sethian and A. Vladimirsky. Ordered upwind methods for static hamilton–jacobi equations. *Proceedings of the National Academy of Sciences*, 98(20):11069–11074, 2001.

[39] Haojie Shi and Max Q. H. Meng. Deep koopman operator with control for nonlinear systems, 2022.

[40] Andrew R. Teel and Laurent Praly. A smooth Lyapunov function from a class-$\mathcal{KL}$ estimate involving two positive semidefinite functions. *ESAIM: Control, Optimisation and Calculus of Variations*, 5:313–367, 2000.

[41] Yen-Hsi Richard Tsai, Li-Tien Cheng, Stanley Osher, and Hong-Kai Zhao. Fast sweeping algorithms for a class of hamilton–jacobi equations. *SIAM Journal on Numerical Analysis*, 41(2):673–694, 2003.

[42] Umesh Vaidya and Prashant G. Mehta. Lyapunov measure for almost everywhere stability. *IEEE Transactions on Automatic Control*, 53(1):307–323, 2008.

[43] A. Vannelli and M. Vidyasagar. Maximal lyapunov functions and domains of attraction for autonomous nonlinear systems. *Automatica*, 21(1):69–80, 1985.

[44] Matthew O. Williams, Ioannis G. Kevrekidis, and Clarence W. Rowley. A data–driven approximation of the koopman operator: Extending dynamic mode decomposition. *Journal of Nonlinear Science*, 25(6):1307–1346, jun 2015.

[45] Guoqiang Yuan and Yinghui Li. Estimation of the regions of attraction for autonomous nonlinear systems. *Transactions of the Institute of Measurement and Control*, 41:106 – 97, 2019.

[46] Ruikun Zhou, Thanin Quartz, Hans De Sterck, and Jun Liu. Neural lyapunov control of unknown nonlinear systems with stability guarantees, 2022.

APPENDICES

Appendix A

Background on Measure Theory

A.1 Primer on Measure spaces

In the Appendix we quickly include the most pressing results on measure theory required to read this thesis.

Definition 11. *Let X be a set. A collection $\mathcal{A}$ of subsets of X is called a σ-algebra of sets in X if the following three statements are true:*

1. *The set X is in $\mathcal{A}$.*
2. *If $A \in \mathcal{A}$, then $A^c \in \mathcal{A}$, where A^c is the complement of A in X.*
3. *If $A_n \in \mathcal{A}$ for all $n \in \mathbb{N}$, and if $A = \bigcup_{n=1}^{\infty} A_n$, then $A \in \mathcal{A}$.*

A set X together with a σ-algebra $\mathcal{A}$ is called a measurable space and is denoted by $(X, \mathcal{A})$. The elements $A \in \mathcal{A}$ are known as measurable sets.

Definition 12. *Let $(X, \mathcal{A})$ be a measurable space. A scalar-valued function f with domain X is called measurable if $f^{-1}(V) \in \mathcal{A}$ whenever V is an open set in the scalar field.*

Definition 13. *Let $(X, \mathcal{A})$ be a measurable space. A set function $\mu : \mathcal{A} \to [0, \infty]$ is said to be countably additive if*

$$\mu\left(\bigcup_{j=1}^{\infty} A_j\right) = \sum_{j=1}^{\infty} \mu\left(A_j\right),$$

whenever $(A_j)_{j=1}^{\infty}$ is a sequence of pairwise disjoint measurable sets. By pairwise disjoint we mean $A_j \cap A_k = \emptyset$whenever $j \neq k$.

A countably additive set function $\mu : \mathcal{A} \to [0, \infty]$ such that $\mu(\emptyset) = 0$ is called a positive measure on $\mathcal{A}$. In such a case, we call the triple $(X, \mathcal{A}, \mu)$ a positive measure space. If the σ-algebra is understood, we often write (X, μ) for the positive measure space and say μ is a positive measure on X.

Example A.1.1. *The Borel σ-algebra on $\mathbb{R}$ is the smallest σ-algebra that contains all of the open intervals in $\mathbb{R}$. A measure defined on the Borel σ-algebra on $\mathbb{R}$ is called a Borel measure on $\mathbb{R}$.*

In the context of probability theory, measurable functions are called random variables and probability measures are simply measures such that $\mu(X) = 1$. Like real valued functions, Borel measures also have notions of convergence. One such is called weak convergence (which should not be mistaken with convergence in the weak* topology, although the two coincide if the underlying space X is compact).

Definition 14. *A sequence of Borel measures μ_i converges weakly to a measure μ if $\lim_{i\to\infty} \int f d\mu_i = \int f d\mu$ for all continuous and bounded f.*

The following are criterion for when a sequence of probability measures has a weakly convergent subsequence. The proof of the theorem can be found in [2].

Definition 15. *A sequence of probability measures $\mathbb{P}_n$ on metric space S is defined to be tight if for every $\epsilon > 0$ there exists n_0 and a compact set $K \subset S$, such that $\mathbb{P}_n(K) > 1 - \epsilon$ for all $n > n_0$.*

Theorem 19. *(Prohorov's Theorem). Suppose sequence $\mathbb{P}_n$ is tight. Then it contains a weakly convergent subsequence $\mathbb{P}_{n(k)} \Rightarrow \mathbb{P}$.*

Remark 12. *Note that by normalization we have that the previous theorem also applies to finite measures.*

A.2 Projections on Measure Spaces

Definition 16 (Hilbert Space)**.** *A Hilbert space $\mathcal{H}$ is a complete normed vector space equipped with an inner product $\langle \cdot, \cdot \rangle$*

An example of a Hilbert space that is crucial to this thesis is $L^2(\mu)$ where μ is a (positive) measure on $\mathcal{M}$. The space $L^2(\mu)$ is defined as the collection of all measurable functions

$\phi : \mathcal{M} \to \mathbb{C}$ satisfying

$$\|\phi\|_{L_2(\mu)} := \sqrt{\int_{\mathcal{M}} |\phi(x)|^2 d\mu(x)} < \infty$$

On this space, the inner product that induces the norm is given by

$$\langle \phi, \psi \rangle := \int_{\mathcal{M}} \phi \bar{\psi} d\mu.$$

Projections on Hilbert spaces are analogous to their finite dimensional counterparts with the exception that subspaces in finite dimensions are automatically closed. The following theorem asserts that for closed subspaces, the projection operator is well defined.

Theorem 20. *If $\mathcal{M}$ is a Hilbert space, K is a closed subspace of $\mathcal{M}$ and $h \in \mathcal{M}$ then there exists a unique point k_0 in K such that*

$$\|h - k_0\| = dist(h, K) := \inf\{\|h - k : k \in K\}.$$

Moreover, $h - k_0 \perp \mathcal{M}$ and the projection operator $P : \mathcal{M} \to K$ is defined to be the point Ph such that $h - Ph \perp \mathcal{M}$.

Assuming then that K is a closed subspace of $L^2(\mu)$, Theorem 20 says that the $L_2(\mu)$-projection of a function $\phi \in L_2(\mu)$ onto $K \subset L_2(\mu)$ is defined by

$$P_N^{\mu}\phi = \arg\min_{f \in K} \|f - \phi\|_{L_2(\mu)} = \arg\min_{f \in K} \int_{\mathcal{M}} |f - \phi|^2 d\mu. \tag{A.1}$$

Appendix B

Proof of Theorem 6

The idea will be to first approximate f by a smooth function and then approximate this smooth function by a neural network. In this end, define $\eta \in C^{\infty}(\mathbb{R}^n)$ by

$$\eta(x) := \begin{cases} C \exp\left(\frac{1}{|x|^2-1}\right) & \text{if } |x| < 1 \\ 0 & \text{if } |x| \geq 1 \end{cases}$$

where the constant C is some normalizing constant, that is $C > 0$ is selected so that $\int_{\mathbb{R}^n} \eta dx = 1$. Some standard properties of $\eta(x)$ is that $\eta \geq 0$, $\eta \in C^{\infty}(\mathbb{R}^n)$ and $\operatorname{spt}(\eta) \subset B_1(0)$ which is the unit ball in $\mathbb{R}^n$. For each $\epsilon > 0$, set

$$\eta_\epsilon(x) := \frac{1}{\epsilon^n} \eta\left(\frac{x}{\epsilon}\right).$$

We call η the standard mollifier. The functions η_ϵ are C^∞ and satisfy

$$\int_{\mathbb{R}^n} \eta_\epsilon dx = 1, \operatorname{spt}(\eta_\epsilon) \subset B(0, \epsilon).$$

Then, by taking the convolution of f with the mollifier η_ϵ, denote $f_\epsilon = f \star \eta_\epsilon$ which can be further simplified to

$$\begin{aligned}f_\epsilon(x) &= \int_{\mathbb{R}^n} f(y)\eta_\epsilon(x-y)dy \\ &= \int_{\mathbb{R}^n} f(x-y)\eta_\epsilon(y)dy \\ &= \frac{1}{\epsilon^n}\int_{B(0,\epsilon)} f(x-y)\eta(\frac{y}{\epsilon}) \\ &= \int_{B(0,1)} f(x-\epsilon y)\eta(y)dy.\end{aligned}$$

It is well known that $f_\epsilon \in C^\infty$. Additionally, by the Lipschitz continuity of f, uniform convergence holds on $\mathbb{R}^n$:

$$\begin{aligned}|f(x)-f_\epsilon(x)| \leq \int_{B(0,1)} |(f(x-\epsilon y)-f(x))\eta_\epsilon(y)|dy \leq L\int_{B(0,1)} \|\epsilon y\|\eta_\epsilon(y)dy \\ \leq L\epsilon \int_{B(0,1)} \eta_\epsilon(y)dy \leq L\epsilon.\end{aligned}$$

However, to define f_ϵ, we need this function to be defined on $\mathbb{R}^n$. Therefore, we need a specific case of the following lemma called the Kirszbraun theorem. The proof of this result can be found in [37].

Lemma 6. *Suppose that $U \subset \mathbb{R}^n$. For any Lipschitz map $f : U \to \mathbb{R}^m$ there exists a Lipschitz-continuous map*

$$F : \mathbb{R}^n \to \mathbb{R}^m$$

that extends f and has the same Lipschitz constant as f.

We are now ready to prove Theorem 6.

Proof. As the proof of (a) is analogous to the proof of (b) and simpler, we elect to prove (b) only. Since f is L-Lipschitz in the uniform norm, we have that the component functions f_i are L-Lipschitz. Since neural networks can be stacked in parallel, it suffices to prove this result for the case that $m = 1$. Moreover, since K is compact, by taking the approximation on some hypercube containing K and then restricting onto K we can also assume without loss of generality that K is convex. The uniform norm is still well-defined as the restriction of a continuous function is continuous. By the Kirszbraun theorem there exists an extension

F to $\mathbb{R}^n$ with the same Lipschitz constant L so we can suppose without loss of generality that f is defined on $\mathbb{R}^n$ and has Lipschitz constant L. Denote $f_\epsilon := f \star \eta_\epsilon$. Since we have that $f_\epsilon(x) = \int_{\mathbb{R}^n} f(x-y)\eta_\epsilon(y)dy$ from the above calculation, we see that

$$\begin{aligned}
|f_\epsilon(x_1) - f_\epsilon(x_2)| &\leq \left| \int_{\mathbb{R}^n} |\left(f(x_1 - y) - f(x_2 - y)\right) \eta_\epsilon(y) dy \right| \\
&\leq \int_{\mathbb{R}^n} |\left(f(x_1 - y) - f(x_2 - y)\right) \eta_\epsilon(y)| \, dy \\
&\leq L\|x_1 - x_2\| \int_{\mathbb{R}^n} \eta_\epsilon(y) dy \\
&= L\|x_1 - x_2\|.
\end{aligned}$$

This shows that f_ϵ is L-Lipschitz. Since $f_\epsilon \to f$ uniformly we can choose $\epsilon > 0$ sufficiently small so that $\|f - f_\epsilon\| < \epsilon/2$. Therefore, by Theorem 3, there exists a neural network ϕ such that $\sup_{x \in K} |f(x) - \phi(x)| < \epsilon/2$ and $\sup_{x\in K} |\frac{\partial f}{\partial x_i}(x) - \frac{\partial \phi}{\partial x_i}(x)| < \epsilon/2$ for all $i = 1, \ldots, n$. Since f is L-Lipschitz it follows that $\|\nabla f\|_2 \leq L$. By the uniform bound on the partial derivatives and the following inequality, $(a+b)^2 \leq (1+\epsilon)a^2 + (1+1/\epsilon)b^2$, this gives

$$\begin{aligned}
\|\nabla \phi\|_2 &= \sqrt{\left(\frac{\partial \phi}{\partial x_1}\right)^2 + \ldots + \left(\frac{\partial \phi}{\partial x_n}\right)^2} \\
&\leq \sqrt{\left(\frac{\partial f}{\partial x_1} \pm \frac{\epsilon}{2}\right)^2 + \ldots + \left(\frac{\partial f}{\partial x_n} \pm \frac{\epsilon}{2}\right)^2} \\
&\leq \sqrt{(1+\epsilon)\left(\left(\frac{\partial f}{\partial x_1}\right)^2 + \ldots + \left(\frac{\partial f}{\partial x_n}\right)^2\right)} + \frac{\sqrt{n}\epsilon}{2}\sqrt{1 + 1/\epsilon} \\
&\leq L + \epsilon\left(\frac{\sqrt{n + n/\epsilon}}{2} + L\right)
\end{aligned}$$

Therefore, by the mean value theorem and the convexity of K, it follows that ϕ is $L + \epsilon\left(\frac{\sqrt{n+n/\epsilon}}{2} + L\right)$-Lipschitz. □