

# Learning Koopman-based Stability Certificates for Unknown Nonlinear Systems

Ruikun Zhou, Yiming Meng, Zhexuan Zeng, and Jun Liu, *Senior Member, IEEE*

**Abstract**—Koopman operator theory has gained significant attention in recent years for identifying discrete-time nonlinear systems by embedding them into an infinite-dimensional linear vector space. However, providing stability guarantees while learning the continuous-time dynamics, especially under conditions of relatively low observation frequency, remains a challenge within the existing Koopman-based learning frameworks. To address this challenge, we propose an algorithmic framework to simultaneously learn the vector field and Lyapunov functions for unknown nonlinear systems, using a limited amount of data sampled across the state space and along the trajectories at a relatively low sampling frequency, potentially as low as  $10Hz$  or less. The proposed framework builds upon recently developed high-accuracy Koopman generator learning for capturing transient system transitions and physics-informed neural networks for training Lyapunov functions. We show that the learned Lyapunov functions can be formally verified using a satisfiability modulo theories (SMT) solver and provide less conservative estimates of the region of attraction compared to existing methods.

**Index Terms**—Koopman operator, stability analysis, Lyapunov functions, operator learning

## I. INTRODUCTION

Stability properties can be qualitatively characterized by Lyapunov functions for various nonlinear systems. However, discovering a Lyapunov function remains a long-standing challenge in nonlinear dynamical systems, even for known systems. For unknown systems, this challenge is further compounded by the accuracy of data-driven system identification and the formal verification of the learned Lyapunov functions. Below, we review the recent advances and challenges associated with these two factors before proposing the learning structure for stability certificates.

The development of Koopman operator theory [9] for dynamical systems provides a promising alternative learning approach for nonlinear system identification and stability analysis [15], using data from snapshots of the flow map (trajectories). By leveraging the spectral properties [19] and mode decomposition capabilities [23], [27] of Koopman operators, nonlinear dynamics can be converted into a discrete

This research was supported in part by an NSERC Discover Grant and the Canada Research Chairs program.

Ruikun Zhou and Jun Liu are with the Department of Applied Mathematics, Faculty of Mathematics, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada. Emails: ruikun.zhou@uwaterloo.ca, j.liu@uwaterloo.ca

Yiming Meng is with the Coordinated Science Laboratory, University of Illinois Urbana-Champaign, Urbana, IL 61801, USA. Email: ymmeng@illinois.edu

Zhexuan Zeng is with Department of Automatic Control, Huazhong University of Science and Technology, Wuhan, China. Email: xuanxuan@hust.edu.cn

family of observable functions (functions of the states), which evolve linearly in the infinite-dimensional function space driven by the Koopman operator. This nonlinear-to-linear conversion, seemingly friendly at first, is later found to require the study of continuous functional calculus. It suffers from restrictive assumptions on the function properties [12] to ensure correct usage.

In identifying the continuous-time dynamics of an unknown system, [13] proposed a method using the system's (infinitesimal) generator, also known as the Koopman generator. This approach avoids the need for time derivatives, requiring fewer data and outperforming widely used methods at low sampling frequencies, such as the sparse identification of nonlinear dynamics (SINDy) technique [2], [8], which heavily relies on low-resolution finite-difference approximations of time derivatives from discrete-time sampled snapshots. However, the method in [13] requires the desirable diagonal property of the Koopman operator, which is typically difficult to satisfy for most unknown nonlinear systems, thus limiting its correct usage. This limitation has been addressed by recent research in [17], which improves learning reliability and accuracy.

Koopman operators also offer a powerful tool for constructing Lyapunov functions and stability analysis [14], [28]. In [14], the authors demonstrated that a set of Lyapunov functions for nonlinear systems with global stability can be constructed from the eigenfunctions of the Koopman operator. Building on this result, and incorporating the autoencoder structure [1] in Koopman operator learning, the authors of [4] introduced an algorithm to identify Koopman eigenfunctions that parameterize a set of Lyapunov function candidates. More recently, [25] used the spectrum of the Koopman operator to directly identify the stability boundary of nonlinear systems. However, due to the low-resolution quantification of the Lyapunov candidate function space, as quantified by the Lyapunov derivative inequality, all the aforementioned stability verification methods face the limitation of conservative estimation of the region of attraction (ROA). In contrast, [16] proposed a modified Zubov-Koopman operator approach to approximate a maximal Lyapunov function, which is related to the solution of stability-related PDEs (Lyapunov's equation or Zubov's equation [11]), where the non-trivial domain corresponds to the ROA. Regardless, all existing Koopman-based constructions of Lyapunov functions cannot be easily integrated into the system identification structure, and therefore lack the ability to be formally verified using the predicted system vector field, leading to a loss of the true predictability of the certifiable region of attraction.

In this paper, we propose an integrated algorithmic framework to simultaneously learn the vector field and Lyapunov functions with formal guarantees for unknown nonlinear systems, thereby achieving full predictability of stability. Our methodology leverages the strengths of the Koopman generator learning framework and utilizes the learned generator to approximately solve stability-related PDEs, with the aim of enhancing the formally verified ROA. In light of the recent development of physics-informed neural network solutions for PDEs, the proposed method in this paper is not merely a straightforward combination of Koopman generator learning and PDE solving using a parameterized ansatz to match the data and the equation. The input of human knowledge on the physics-informed conditions and the corresponding design of the loss function to train the solution are essential components.

It is worth noting that the technique proposed in [16] can learn a near-maximal Lyapunov candidate for unknown systems, but requires a computationally intensive deep neural network smoothing process. Moreover, the Lyapunov derivative of the candidate cannot be verified due to the lack of information about the system's vector field. On the other hand, a non-Koopman neural system learning and Lyapunov construction framework [30] can provide formal guarantees to ensure closed-loop stability. However, this approach does not offer insights into the quality of the resulting ROA. We address these shortcomings and, to the best of the authors' knowledge, this is the first work to develop a systematic approach for finding formally verified Lyapunov functions using the Koopman generator.

The detailed contributions are as follows.

- We propose a streamlined framework to simultaneously learn the vector field and Lyapunov functions by reusing the same observable test functions and a subset of the data samples.
- Beyond formulating the least-squares problem, we provide additional conditions for solving the stability-related PDEs using the Koopman generator.
- We demonstrate that the learned Lyapunov function is valid for unknown systems, with formal guarantees achieved through verification using SMT solvers.

## II. PRELIMINARIES

Throughout this paper, we consider a continuous-time nonlinear dynamical system of the form

$$\dot{x} = f(x), \quad x(0) = x_0, \quad (1)$$

and the vector field  $f : \mathcal{X} \rightarrow \mathbb{R}^n$  is assumed to be locally Lipschitz continuous, where  $\mathcal{X} \subseteq \mathbb{R}^n$  is a pre-compact state space. For each initial condition  $x_0$ , we denote the unique forward flow map, i.e., the solution map, by  $\phi : I \times \mathcal{X} \rightarrow \mathcal{X}$ , where  $I$  is the maximal interval of existence. In this paper, we assume  $I = [0, \infty)$  for the initial value problem (1). The flow map should then satisfy 1)  $\partial_t(\phi(t, x)) = f(\phi(t, x))$ , 2)  $\phi(0, x) = x_0$ , and 3)  $\phi(s, \phi(t, x)) = \phi(t + s, x)$  for all  $t, s \in I$ .

## A. Koopman Operators and the Infinitesimal Generator

**Definition 2.1:** The Koopman operator family  $\{\mathcal{K}_t\}_{t \geq 0}$  of system (1) is a collection of maps  $\mathcal{K}_t : \mathcal{C}^1(\mathcal{X}) \rightarrow \mathcal{C}^1(\mathcal{X})$  defined by

$$\mathcal{K}_t h = h \circ \phi(t, \cdot), \quad h \in \mathcal{C}^1(\mathcal{X}) \quad (2)$$

for each  $t \geq 0$ , where  $\circ$  is the composition operator. The (infinitesimal) generator  $\mathcal{L}$  of  $\{\mathcal{K}_t\}_{t \geq 0}$ , which is also called the Koopman generator, is defined as  $\mathcal{L}h := \lim_{t \rightarrow 0} \frac{\mathcal{K}_t h - h}{t}$  and equivalent to

$$\mathcal{L}h(x) = \nabla h(x) \cdot f(x), \quad h \in \mathcal{C}^1(\mathcal{X}). \quad (3)$$

It is known that  $\{\mathcal{K}_t\}_{t \geq 0}$  forms a linear  $\mathcal{C}_0$ -semigroup [18], and thus there exist constants  $\omega \geq 0$  (indicating the exponential growth rate) and  $C \geq 1$  (representing a spatial uniform scaling) such that  $\|\mathcal{K}_t\| \leq C e^{\omega t}$ ,  $\forall t \geq 0$  [20].

## B. Concept of Stability

Without loss of generality, we assume the origin is an equilibrium point of the system (1).

**Definition 2.2 (Asymptotic Stability):** The origin is said to be asymptotically stable for (1) if 1) for every  $\varepsilon > 0$ , there exists a  $\delta > 0$  such that  $|x| < \delta$  implies  $|\phi(t, x)| < \varepsilon$  for all  $t \geq 0$ , and 2) there exists a neighborhood  $U \subseteq \mathcal{X}$  of the origin such that, for each  $x_0 \in U$ , we have  $\phi(t, x) \in U$  for all  $t \geq 0$  and  $\lim_{t \rightarrow \infty} |\phi(t, x)| = 0$ .

We further define the region of attraction of the equilibrium point given its asymptotic stability.

**Definition 2.3 (ROA):** Suppose that the origin is asymptotically stable. The domain of attraction of it is a set defined as  $\mathcal{D} := \{x \in \mathcal{X} : \lim_{t \rightarrow \infty} |\phi(t, x)| = 0\}$ . Any forward invariant subset  $\mathcal{R} \subseteq \mathcal{D}$  is called a region of attraction (ROA).

**Theorem 2.4 (Lyapunov Theorem):** Let  $D \subseteq \mathbb{R}^n$  be an open set containing the origin. Consider the nonlinear system (1). Let  $V(x) : D \rightarrow \mathbb{R}$  be a continuously differentiable function such that

$$\begin{aligned} V(\mathbf{0}) &= 0 \text{ and } V(x) > 0 \text{ for all } x \in D \setminus \{\mathbf{0}\}, \\ \mathcal{L}V(x) &:= \nabla V(x) \cdot f(x) < 0 \text{ for all } x \in D \setminus \{\mathbf{0}\}, \end{aligned} \quad (4)$$

Then, the origin is asymptotically stable for the system.

An immediate consequence of Theorem 2.4 (and its proof [7]) is that the sub-level set  $V^c = \{x \in D \mid V(x) \leq c\}$  is an ROA for the system (1), where  $c > 0$  is such that  $V^c$  does not intersect with the boundary of  $D$ .

Given that the origin of a nonlinear system is asymptotically stable, a Lyapunov function exists by the converse Lyapunov theorem [7]. But converse Lyapunov functions are usually nonconstructive. Despite their fundamental importance, computing Lyapunov functions remains a challenging task. Existing methods, such as sum-of-squares (SOS) and neural Lyapunov functions, typically yield local results with highly conservative estimates of the ROA, as their searching policies are guided by the Lyapunov derivative inequality, which admits non-unique solutions. In contrast, as demonstrated in [11], [29] and the recent literature, the maximal Lyapunov function [26], which provides the largest

estimation of ROA, can be characterized by the following partial differential equation (PDE):

$$\mathcal{L}V(x) = -\eta(x), \quad (5)$$

which we refer to as Lyapunov's equation. Here  $\eta$  is a positive definite function over  $\mathcal{X}$  with respect to the origin. However, due to the unbounded nature of the Lyapunov function, it remains challenging to approximate the true domain of attraction using typical data-driven learning methods, given the compactness of the sampling subspace. The following theorem states that the domain of attraction can be characterized by a transformed maximal Lyapunov function with a bounded range from 0 to 1, where the 1-sublevel set represents the domain of attraction.

**Theorem 2.5 (Zubov's Theorem [31]):** Let  $D \subset \mathbb{R}^n$  be an open set containing the origin. Then  $D = \mathcal{D}$  if and only if there exist two continuous functions  $W : D \rightarrow \mathbb{R}$  and  $\eta : D \rightarrow \mathbb{R}$  such that the following conditions hold: 1)  $0 < W(x) < 1$  for all  $x \in D \setminus \{\mathbf{0}\}$  and  $W(\mathbf{0}) = 0$ ; 2)  $\eta$  is positive definite on  $D$  with respect to 0; 3) for any sufficiently small  $c_3 > 0$ , there exist  $c_1, c_2 > 0$  such that  $|x| \geq c_3$  implies  $W(x) > c_1$  and  $\eta(x) > c_2$ ; 4)  $W(x) \rightarrow 1$  as  $x \rightarrow y$  for any  $y \in \partial D$ ; 5)  $W$  and  $\eta$  satisfy

$$\mathcal{L}W(x) + \eta(x)(1 - W(x)) = 0. \quad (6)$$

We refer to (6) as Zubov's equation in the following context.

### C. Problem Formulation

The idea of this paper is to identify the Koopman generator and then solve the stability-related PDEs outlined above, with the goal of enlarging the estimation of the ROA. In previous work, [11], [29] demonstrated excellent performance in learning accuracy. However, when solving PDEs with learned generators for unknown systems, a simple data-fitting strategy by minimizing the residual is typically insufficient, as additional physical information—such as boundary conditions—must also be incorporated. In the following section, we first recap the resolvent-type method for learning the Koopman generator in [18]. In Section V, we focus on developing the physics-informed Koopman-generator-based construction of the Lyapunov function to address the technical concerns mentioned above.

### III. LEARNING THE GENERATOR AND IDENTIFYING THE VECTOR FIELD VIA KOOPMAN RESOLVENT

In this section, we recap the approximation of the Koopman generator and show that the generator can be approximated by the Koopman resolvent [24].

For  $h \in \mathcal{C}^1(\mathcal{X})$ , we define the following integral representation of the Koopman resolvent:  $\mathcal{R}(\lambda; \mathcal{L})h := \int_0^\infty e^{-\lambda t} (\mathcal{K}_t h) dt$ . Consequently, the Yosida approximation of the Koopman generator  $\mathcal{L}$  can be defined as  $\mathcal{L}_\lambda = \lambda^2 \mathcal{R}(\lambda; \mathcal{L}) - \lambda \text{Id}$ , where  $\text{Id}$  is the identity operator [20]. We have that  $\mathcal{L}_\lambda$  converges to  $\mathcal{L}$  strongly as  $\lambda \rightarrow \infty$ . In order to have a finite time approximation, we define a truncation integral operator as

$$\mathcal{R}_{\lambda, \tau_s} h(x) := \int_0^{\tau_s} e^{-\lambda s} \mathcal{K}_s h(x) ds. \quad (7)$$

Then, when  $\lambda$  is sufficiently large, we should have an accurate approximation of the Koopman generator, as stated in the following theorem.

**Theorem 3.1 ([18]):** Define  $\mathcal{L}_{\lambda, \tau_s} := \lambda^2 \mathcal{R}_{\lambda, \tau_s} - \lambda \text{Id}$ , and let  $\tau_s \geq 0$  and  $\lambda > \omega$  be fixed. Then,  $\|\mathcal{L}_{\lambda, \tau_s} - \mathcal{L}\| \leq \frac{C\lambda^2}{\lambda - \omega} e^{-\lambda \tau_s}$  on  $\mathcal{C}^1(\mathcal{X})$ .

We refer the reader to [18] for detailed derivation and proofs. However, when  $\lambda$  is large, it poses a challenge as it requires high sampling frequencies in practical applications. We then leverage the first resolvent identity  $[(\lambda - \mu)\mathcal{R}(\mu; \mathcal{L}) + \text{Id}]\mathcal{R}(\lambda; \mathcal{L}) = \mathcal{R}(\mu; \mathcal{L})$  to tackle this issue. By computing a finite-time horizon approximation  $\mathcal{R}_{\mu, \tau_s}$  for  $\mathcal{R}(\mu; \mathcal{L})$  first,  $\mathcal{L}_{\lambda, \tau_s}$  can be computed using this identity, where  $\mu$  is a small positive number in the resolvent set of the Koopman generator.

By (3) with  $x := [x_1, x_2, \dots, x_n]^T \in \mathbb{R}^n$ , we have  $\mathcal{L}x_i = f_i(x)$ , for  $i = 1, 2, \dots, n$ , where  $f_i$  is defined as  $\dot{x}_i = f_i(x)$ , a component of  $f$ . As a result of Theorem 3.1, when  $x_1, x_2, \dots, x_n$  are included in the observable test functions, the vector field can be readily approximated:

$$\hat{f}_i(x) = \mathcal{L}_{\lambda, \tau_s} x_i \approx \mathcal{L}x_i = f_i(x), \quad \forall i = 1, 2, \dots, n. \quad (8)$$

We also direct the readers to [18] to see a set of numerical results on the effectiveness of (8) on the system identification tasks, varying from polynomial systems to general nonlinear systems.

### IV. STABILITY CERTIFICATES FOR THE UNKNOWN SYSTEMS

In this section, we show that the Koopman generator can be used not only to approximate the vector field but also to establish stability certificates for unknown systems, i.e., constructing Lyapunov functions by solving linear PDEs. Furthermore, we show that the derived Lyapunov functions provide stability guarantees for the unknown systems.

We assume that the origin is a stable equilibrium point of (1). Then (1) can be rewritten as  $\dot{x} = Ax + g(x)$ , where  $g(x) = f(x) - Ax$  satisfies  $\lim_{x \rightarrow \mathbf{0}} \frac{\|g(x)\|}{\|x\|} = 0$ . If  $A$  is known, a local quadratic Lyapunov function  $V_P(x) = x^T P x$  can be computed by solving  $PA + A^T P = -Q$ , where  $Q$  is any symmetric positive definite matrix. It can be easily shown that there exists a sufficiently small  $c > 0$  such that  $\mathcal{O} = \{x \in \mathcal{X} | V_P(x) \leq c\}$ , where the linearization dominates, that is an ROA [11, Section 5.1]. We make the following assumption.

**Assumption 4.1:** Let the unknown system be identified using (8) and denote the linearization of the approximated vector field about the origin as  $\dot{x} = \hat{A}x$ . With  $\hat{A}$  being Hurwitz (which can be checked numerically), let  $P$  solve the Lyapunov equation  $P\hat{A} + \hat{A}^T P = -Q$  for some positive definite  $Q$ . We assume that identification is sufficiently accurate such that  $\mathcal{O} = \{x \in \mathcal{X} | V_P(x) \leq c\}$  is an ROA for (1) for some  $c > 0$ .

Note that this assumption can typically be satisfied in most cases by collecting a sufficient amount of data around the origin and ensuring  $f(\mathbf{0}) = \hat{f}(\mathbf{0})$  in practice. Now, consider an unsampled state  $z$ , and  $y$  represents the nearest

known sample for which  $f(y)$  and  $\hat{f}(y)$  are accessible. The following theorem establishes the stability conditions for the unknown systems.

**Proposition 4.2:** Suppose that Assumption 4.1 holds. Let  $V : \mathcal{X} \rightarrow \mathbb{R}$  be a candidate Lyapunov function for  $\dot{x} = \hat{f}(x)$ , with  $\hat{f}(\mathbf{0}) = 0$ , and let  $K_f$  and  $K_{\hat{f}}$  be the Lipschitz constants of  $f$  and  $\hat{f}$  respectively, on  $\mathcal{X}$ . Let  $\mathcal{Y} \subset \mathcal{X}$  be a finite set of samples. Let  $\delta$ ,  $\nu$ , and  $\alpha$  be positive constants such that 1) for every  $z \in \mathcal{X}$ , there exists  $y \in \mathcal{Y}$  such that  $\|z - y\| < \delta$ , 2)  $\sup_{x \in \mathcal{X}} \|\nabla V(x)\| \leq \nu$ , and 3)  $\max_{y \in \mathcal{Y}} \|f(y) - \hat{f}(y)\| = \alpha$ . Let  $\beta$  be such that  $((K_f + K_{\hat{f}})\delta + \alpha)\nu < \beta$  and

$$\nabla V(x) \cdot \hat{f}(x) \leq -\beta \quad (9)$$

holds on  $\Omega_{c_1, c_2} = \{x \in \mathcal{X} : c_1 \leq V(x) \leq c_2\}$ , for some  $0 < c_1 < c_2$ , such that  $\Omega_{c_1} = \{x \in \mathcal{X} : V(x) \leq c_1\} \subset \mathcal{O}$ , then  $\Omega_{c_2} = \{x \in \mathcal{X} : V(x) \leq c_2\}$  is an ROA for the unknown system (1), provided that  $\Omega_{c_2}$  does not intersect with the boundary of  $\mathcal{X}$ .

*Proof:* For any  $x \in \mathcal{X}$  and  $y \in \mathcal{Y}$ , we have

$$\begin{aligned} & \|f(x) - \hat{f}(x)\| \\ & \leq \|f(x) - f(y)\| + \|f(y) - \hat{f}(y)\| + \|\hat{f}(y) - \hat{f}(x)\| \\ & \leq K_f \delta + \alpha + K_{\hat{f}} \delta. \end{aligned} \quad (10)$$

It follows that

$$\begin{aligned} & \nabla V(x) \cdot f(x) - \nabla V(x) \cdot \hat{f}(x) \\ & \leq \|\nabla V\| \|f(x) - \hat{f}(x)\| \\ & \leq \nu((K_f + K_{\hat{f}})\delta + \alpha) < \beta. \end{aligned} \quad (11)$$

By (9) and (11), we have  $\nabla V(x) \cdot f(x) < \nabla V(x) \cdot \hat{f}(x) + \beta < 0$  on  $\Omega_{c_1, c_2}$ . Hence all the trajectories in  $\Omega_{c_2}$  converge to  $\Omega_{c_1} \subset \mathcal{O}$ . By Assumption 4.1, solutions of (1) will converge to the origin from  $\mathcal{O}$ . ■

## V. DATA-DRIVEN ALGORITHM FOR DERIVING STABILITY CERTIFICATES

In this section, we continue to discuss the data-driven learning algorithm based on the derivation for learning the vector field and Lyapunov function in the previous two sections.

### A. Learning the Koopman Generator

First, we select a finite dictionary of continuously differentiable observable test functions, denoted by  $\mathcal{Z}_N(x) := [\mathfrak{z}_1(x), \mathfrak{z}_2(x), \dots, \mathfrak{z}_N(x)]$ ,  $N \in \mathbb{N}$ . Suppose that there exists a data-driven finite-dimensional approximation of  $\mathcal{L}_{\lambda, \tau_s}$ , denoted as  $L$ , such that for any  $h \in \text{span}\{\mathfrak{z}_1, \mathfrak{z}_2, \dots, \mathfrak{z}_N\}$ , i.e.,  $h(x) = \mathcal{Z}_N(x)\zeta$  where  $\zeta$  is a column vector, we have that  $\mathcal{L}h(\cdot) \approx \mathcal{L}_{\lambda, \tau_s} h(\cdot) \approx \mathcal{Z}_N(\cdot)(L\zeta)$ .

By randomly sampling  $M$  initial conditions  $\{x^{(m)}\}_{m=1}^M \subset \mathcal{X}$  and fixing a  $\tau_s$  with a sampling rate of  $\gamma$  Hz along the trajectories, we construct the observable functions into  $X$ , as  $B = [\mathcal{Z}_N(x^{(1)}), \mathcal{Z}_N(x^{(2)}), \dots, \mathcal{Z}_N(x^{(M)})]^T$ . Under the specified sampling rate, to avoid inaccurate computation of the truncated integral of the Koopman resolvent (7) for large

$\lambda$ , we first apply the Gauss–Legendre quadrature method [21] using trajectory samples to obtain the numerical approximation  $\hat{\mathcal{R}}_{\mu, \tau_s}$  for  $\mathcal{R}_{\mu, \tau_s}$  with a relatively small constant  $\mu > 0$ . Subsequently, the generator  $L$  can be inferred as

$$L = X^\dagger Y \quad (12)$$

where  $X = (\lambda - \mu)\hat{\mathcal{R}}_{\mu, \tau_s} + B$ , and  $Y = \lambda\mu\hat{\mathcal{R}}_{\mu, \tau_s} - \lambda B$ . We direct the readers to [18] for a detailed algorithm and explanation of this resolvent-type method.

By including  $x$  in the dictionary, the approximated vector field  $\hat{f}(x) \approx B\mathcal{Z}_N(x)^T$ , for some  $B \in \mathbb{R}^{n \times N}$ . To ensure that the approximated dynamics preserve the same equilibrium point at the origin, we can perform  $\tilde{f}(x) = \hat{f}(x) - \hat{f}(\mathbf{0})$ , as demonstrated in [22]. In doing so, Assumption 4.1 is easier to be satisfied with  $\tilde{f}(\mathbf{0}) = \mathbf{0}$ . Again, we would like to emphasize that with this proposed method, the vector field is learned via learning the Koopman generator using the discrete sampled points along the trajectories. Namely, one does not need to approximate the time derivatives and then identify the continuous-time systems. Moreover, it contributes to a more accurate identification, in comparison with finite-difference based methods, for instance, SINDy [8], [18].

### B. Learning the Lyapunov Function

We consider a Lyapunov function candidate of the form  $V(x) = \mathcal{Z}_N(x)\theta$  with  $\theta \in \mathbb{R}^N$ . Then, to solve Lyapunov's equation (5) and Zubov's equation (6) respectively, we need the Lyapunov function candidate to satisfy:

$$\mathcal{Z}_N(x)L\theta = -\eta(x), \quad (13)$$

and

$$[\mathcal{Z}_N(x)L - \eta(x)\mathcal{Z}_N(x)]\theta = -\eta(x), \quad (14)$$

respectively. Then, it appears that the problem is reduced to finding the vector  $\theta$  such that the least-squares error between the l.h.s. and r.h.s. of (13) or (14) is minimized at the sampled initial conditions. However, directly solving the linear PDEs in such a way may not return the true solutions, especially for Zubov's equation (14). For instance, it is evident that  $W(x) = 1$  is a trivial solution to (6).

Next, we select a set of boundary points  $\{y^{(p)}\}_{p=0}^{P-1}$  and formulate the least-squares problem for solving Zubov's equation as:

$$\begin{aligned} \theta = \arg \min_{K \in \mathbb{R}^N} & \frac{1}{M} \sum_{i=1}^M \left\| [\mathcal{Z}_N(x_i)L - \eta(x_i)\mathcal{Z}_N(x_i)] K + \right. \\ & \left. \eta(x_i) \right\|^2 + \lambda_b \frac{1}{P} \sum_{i=1}^P \left\| \mathcal{Z}_N(y_i)K - b(y_i) \right\|^2, \end{aligned} \quad (15)$$

where  $\lambda_b > 0$  is a weight parameter, and  $b(y_i)$  is boundary values for the sampled  $y_i$ . In a similar vein, solving the

Lyapunov's equation (13) can be formulated as:

$$\theta = \arg \min_{K \in \mathbb{R}^N} \frac{1}{M} \sum_{i=1}^M \|\mathcal{Z}_N(x_i)LK + \eta(x_i)\|^2 + \lambda_b \frac{1}{P} \sum_{i=1}^P \|\mathcal{Z}_N(y_i)K - b(y_i)\|^2. \quad (16)$$

For the positive definite function  $\eta$ , one common choice is  $\eta(x) = r|x|^2$ , where  $r$  is a positive constant. Unless otherwise specified, we will use it with  $r = 0.1$  in all the following numerical experiments. The boundary condition for Zubov's equation consists of both  $W(\mathbf{0}) = 0$  and  $W(y) = 1$  for  $y \notin \mathcal{D}$  (and  $y \in \partial\mathcal{D}$  if the knowledge is available), whereas the one for Lyapunov's equation is a single point  $V(\mathbf{0}) = 0$ . Note that the positive definiteness of the Lyapunov function  $V$  can be ensured by solving the PDEs accurately with the chosen  $\eta$  [11].

**Remark 5.1:** Note that with the identified vector field  $\tilde{f}(x)$ , the stability certificates can also be attained by directly solving Lyapunov's equation  $\nabla V(x; \theta) \cdot \tilde{f}(x) = -\eta(x)$  and Zubov's equation  $\nabla W(x; \theta) \cdot \tilde{f}(x) = -\eta(x)(1 - W(x; \theta))$ , using the method proposed in [11]. However, the proposed method provides a more efficient approach, requiring only the solution of a least-squares problem by repeatedly using the same (observable) test functions.

The process of identifying the dynamics and computing the Lyapunov functions is summarized in Algorithm 1.

---

**Algorithm 1:** Koopman generator-based Lyapunov function and system identification

---

**Input:**  $X, Y$

**Output:**  $V$  or  $\hat{f} \& V$

- 1 Learn  $L$  using (12);
  - 2 **if** *Identifying the dynamics is needed* **then**
  - 3     Identify the dynamics  $\hat{f}$  with  $L$  using (8);
  - 4     Learn the Lyapunov function  $V$  with (15) or (16);
  - 5 **end**
  - 6 **else**
  - 7     Learn the Lyapunov function  $V$  with (15) or (16);
  - 8 **end**
- 

### C. The Choice of the Dictionary

In this work, we primarily employ two types of observable test functions widely recognized in the literature [13], [18]: monomials and shallow neural networks, featuring randomly initialized weights and biases.

It is well-known that one-hidden layer neural networks are able to approximate any continuous function on a compact set to any desired degree of accuracy [3]. More importantly, as illustrated in [6], [29], the extreme learning machine (ELM) algorithm can solve the linear stability-related PDEs well using single-hidden layer feedforward neural networks. It returns the Lyapunov functions of form  $V(x; \theta) = \sigma(\omega x + b)\theta$ , where  $\omega$  and  $b$  are randomly generated

weights and bias, while  $\theta$  is learned by performing a least-squares task. In this paper, we mainly consider the hyperbolic tangent function  $\tanh$  as the activation function  $\sigma$ . That said, in order to simultaneously learn the vector field and the Lyapunov function via the Koopman generator, we set  $\mathcal{Z}_N = [\tanh(\omega x + b)^T; x]$ , where  $\omega \in \mathbb{R}^{(N-n) \times n}$ ,  $b \in \mathbb{R}^{N-n}$ . Consequently, we have an approximation of the vector field:  $\tilde{f}(x) = \Xi[\tanh(\omega x + b)^T; x]^T$  with  $\Xi = L[(N-n) : \text{end}, :]$ , and  $V(x) = [\tanh(\omega x + b)^T; x]\theta$ .

The above selection of observable test functions applies to general nonlinear systems. However, with specific information about the systems, tailored test functions can improve approximation accuracy. For example, in polynomial systems, monomials can be used for more accurate results. Note that with the information, we potentially can get a better approximation of the Koopman generator, and thus better ROA estimates can be attained. In general, the more accurate the resulting generator and corresponding identified vector field are, the closer the ROA estimate can be to the actual ROA.

## VI. NUMERICAL EXPERIMENTS

We present two numerical examples to demonstrate the effectiveness of the proposed method. For both examples, we assume that the systems are unknown and set  $Q = I$  when solving for a quadratic Lyapunov function  $V_P$ , as shown in Section IV. Recalling equations (13) and (14), Zubov's equation is in general more challenging to solve and yields larger ROA estimates. Hence we focus exclusively on its results in this section. The Lyapunov's equation is a simpler case in terms of solving the linear PDE and can be readily computed by simplifying the process, mainly the residual term and the boundary conditions, with the provided code. The code is available at [https://github.com/RuikunZhou/Unknown\\_Zubov\\_Koopman](https://github.com/RuikunZhou/Unknown_Zubov_Koopman).

### A. Reversed Van der Pol Oscillator

Consider the reversed Van der Pol oscillator  $\dot{x}_1 = -x_2$ ,  $\dot{x}_2 = x_1 - (1 - x_2^2)x_2$ , with  $x := [x_1, x_2]$ . Since this is a polynomial system, we utilize the monomials as the  $N = J \times K$  basis functions with  $\mathfrak{z}_i(x) = x_1^p x_2^q$ ,  $p = \text{mod}(i/J)$ , and  $q = \lfloor i/K \rfloor$ . In order to solve Zubov's equation (14) sufficiently well using polynomials, we choose  $J = K = 8$ . In this case, for generator learning, we only need to randomly sample  $M = 100$  initial conditions on  $\mathcal{X} = [-1.2, 1.2]^2$  and the data points along the trajectories with a sampling frequency  $\gamma = 50\text{Hz}$  for  $\tau_s = 5s$ . In comparison, the feedforward neural network-based method in [30] needs 9 million samples with the exact values of the corresponding time derivative for each data point. As a natural result of the generator approximation  $L$ , we have the identified vector field  $\tilde{f}$ . Also, we verified that Assumption 4.1 indeed holds in this case.

Next, we randomly sample 3000 data points across  $[-2.5, 2.5] \times [-3.5, 3.5]$  with 100 boundary points. Note that the boundary conditions given here do not need to be exactly on  $\partial\mathcal{D}$ , and the samples for generator learning can be reused

here. For this example, we directly use the boundary points on the box which contains  $\mathcal{D}$  as the boundary points for solving the PDE. We then solve Zubov's equation (14) using the learned  $L$  with the help of the well-developed toolbox for finding and verifying Lyapunov functions, LyZNet [10], resulting in a polynomial Lyapunov function. It is remarkable that, differing from the regular Lyapunov conditions defined in Theorem 2.4, we verify the modified Lyapunov conditions for the Lie derivative as (9), using the integrated satisfiability modulo theories (SMT) solvers in the toolbox. The resulting Lyapunov function and the corresponding ROA are included in Fig. 1.

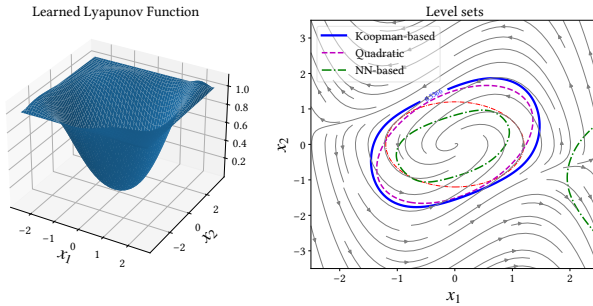


Fig. 1. The learned Lyapunov function and corresponding certified ROA estimates, where the blue curve is the one using the proposed method. The magenta dashed one is the verified largest ROA estimate with  $V_P(x)$ , while the green dot-dashed line is with the neural network-based approach proposed in [30]. The red dot-dashed circle denotes  $\mathcal{X}$  on which we collect the data for computing the Koopman generator.

The parameters for learning the generators and verification for stability are detailed in Table I, where  $\mu$  and  $\lambda$  are used to compute (12). We assume that the Lipschitz constant  $K_f$  is known. For  $\hat{f}$  and  $V$ , we have the exact expressions, given that they are linear combinations of the observable test functions. We thus can leverage symbolic computations and then compute  $K_{\hat{f}}$  and  $\nu$  with the help of the SMT solver, dReal [5], by performing interval analysis with the expressions. Note that when the 2-norm needs to be evaluated, it is over-approximated by the Frobenius norm for the expressions. It is noteworthy that while data is sampled on  $\mathcal{X}$  to learn the dynamics within the domain of attraction, the certified ROA extends beyond this so-called valid region described in [30]. This extension is made possible by incorporating physical insights—specifically, leveraging the knowledge that the system is polynomial. With that, we have a more accurate approximation with improved generalization performance outside  $\mathcal{X}$ . More importantly, both the system identification and stability certification processes require significantly less computational time compared to existing methods. The accuracy and efficiency also result in a substantially smaller  $\beta$  in (9), reduced by a factor of  $\frac{1}{10}$  for verification. This reduction is critically important for practical applications involving real-world systems.

### B. Two-machine Power System

Consider the two-machine power system in [26] with the following governing equation:  $\dot{x}_1 = x_2$ ,  $\dot{x}_2 = -0.5x_2 -$

$(\sin(x_1 + a) - \sin(a))$ , where  $a = \frac{\pi}{3}$ . In this example, the system is non-polynomial with a trigonometric function, and we assume no prior knowledge of its dynamics. The dictionary is constructed using 100 tanh-activated neural networks and the state variable  $x$ . We sample  $M = 50^2$  initial conditions within  $\mathcal{X} = [-1, 1]^2$ , which lies entirely within the true domain of attraction. Following the same procedure as in the previous example, we solve the Zubov's equation over  $[-2, 3] \times [-3, 1.5]$ . The parameter settings are summarized in Table II, and the resulting Lyapunov function and corresponding ROA estimate can be found in Fig. 2.

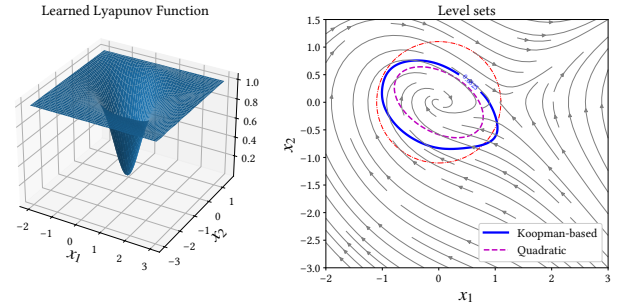


Fig. 2. The learned Lyapunov function and corresponding certified ROA estimates, where the red dot-dashed circle denotes  $\mathcal{X}$  on which we collect the data for computing the Koopman generator. The blue curve is the certified ROA estimate using the proposed method, while the magenta dashed line is the one with  $V_P(x)$ .

We emphasize that the Koopman generator-based method proposed in this work serves as a modular framework for identifying continuous-time systems and finding Lyapunov functions. After identifying the system, one can also directly solve the PDEs to find Lyapunov functions, as discussed in Remark 5.1. On the other hand, if the vector field is identified using other techniques, such as SINDy, the learned generator  $L$  is exclusively used for solving the stability-related PDEs. Notably, leveraging the generator for both system identification and stability analysis significantly reduces computational overhead, underscoring the efficiency of the proposed approach.

## VII. CONCLUSION

In this paper, we propose a framework for simultaneously identifying the vector field and finding a Lyapunov function for an unknown nonlinear system by computing the Koopman generator. Using a shared dictionary of observable test functions for both computing the Koopman generators and solving stability-related PDEs, we establish a more efficient way for constructing stability certificates from data. Leveraging SMT solvers further allows for less conservative ROA estimates, compared to the quadratic Lyapunov functions and an existing neural network-based method. While the current numerical results focus on low-dimensional systems, future work will explore scalability to high-dimensional dynamics with advanced verification tools.

TABLE I

PARAMETERS IN THE VAN DER POL OSCILLATOR CASE, WHERE THE FIRST FOUR PERTAIN TO LEARNING THE KOOPMAN GENERATOR, WHILE THE LAST SIX CORRESPOND TO THOSE IN PROPOSITION 4.2

$\tau_s(s)$	$\gamma$	$\mu$	$\lambda$	$\lambda_b$	$K_f$	$K_{\hat{f}}$	$\delta$	$\alpha$	$\nu$	$\beta$
5	50	2.5	1e8	100	4.90	4.90	3e-4	4.16e-6	7.07e-1	2.08e-3

TABLE II

PARAMETERS FOR THE TWO-MACHINE POWER SYSTEM USING  $\tanh$ -ACTIVATED NEURAL NETWORKS

$\tau_s(s)$	$\gamma$	$\mu$	$\lambda$	$\lambda_b$	$K_f$	$K_{\hat{f}}$	$\delta$	$\alpha$	$\nu$	$\beta$
5	10	3	1e8	100	1.52	1.52	1e-4	2.72e-4	1.45	8.34e-4

## REFERENCES

- [1] Omri Azencot, N Benjamin Erichson, Vanessa Lin, and Michael Mahoney. Forecasting sequential data using consistent koopman autoencoders. In *International Conference on Machine Learning*, pages 475–485. PMLR, 2020.
- [2] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the National Academy of Sciences*, 113(15):3932–3937, 2016.
- [3] George Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- [4] Shankar A Deka, Alonso M Valle, and Claire J Tomlin. Koopman-based neural Lyapunov functions for general attractors. In *61st Conference on Decision and Control (CDC)*, pages 5123–5128. IEEE, 2022.
- [5] Sicun Gao, Soonho Kong, and Edmund M Clarke. dReal: An SMT Solver for Nonlinear Theories over the Reals. In *International conference on automated deduction*, pages 208–214. Springer, 2013.
- [6] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1-3):489–501, 2006.
- [7] H.K. Khalil. *Nonlinear Control*. Always Learning. Pearson, 2015.
- [8] Stefan Klus, Felix Nüske, Sebastian Peitz, Jan-Hendrik Niemann, Cecilia Clementi, and Christof Schütte. Data-driven approximation of the Koopman generator: Model reduction, system identification, and control. *Physica D: Nonlinear Phenomena*, 406:132416, 2020.
- [9] Bernard O Koopman. Hamiltonian systems and transformation in Hilbert space. *Proceedings of the National Academy of Sciences*, 17(5):315–318, 1931.
- [10] Jun Liu, Yiming Meng, Maxwell Fitzsimmons, and Ruikun Zhou. Tool LyZNet: A lightweight Python tool for learning and verifying neural Lyapunov functions and regions of attraction. In *Proceedings of the 27th ACM International Conference on Hybrid Systems: Computation and Control*, pages 1–8, 2024.
- [11] Jun Liu, Yiming Meng, Maxwell Fitzsimmons, and Ruikun Zhou. Physics-informed neural network Lyapunov functions: PDE characterization, learning, and verification. *Automatica*, 175:112193, 2025.
- [12] Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Deep learning for universal linear embeddings of nonlinear dynamics. *Nature Communications*, 9(1):4950, December 2018.
- [13] Alexandre Mauroy and Jorge Goncalves. Koopman-based lifting techniques for nonlinear systems identification. *IEEE Transactions on Automatic Control*, 65(6):2550–2565, 2019.
- [14] Alexandre Mauroy and Igor Mezic. Global Stability Analysis Using the Eigenfunctions of the Koopman Operator. *IEEE Transactions on Automatic Control*, 61(11):3356–3369, November 2016.
- [15] Alexandre Mauroy, Y Susuki, and Igor Mezic. *Koopman operator in systems and control*, volume 484. Springer, 2020.
- [16] Yiming Meng, Ruikun Zhou, and Jun Liu. Learning regions of attraction in unknown dynamical systems via Zubov-Koopman lifting: Regularities and convergence. *arXiv preprint arXiv:2311.15119*, 2023.
- [17] Yiming Meng, Ruikun Zhou, Melkior Ornik, and Jun Liu. Koopman-based learning of infinitesimal generators without operator logarithm. In *The 63rd IEEE Conference on Decision and Control (CDC)*. IEEE, 2024.
- [18] Yiming Meng, Ruikun Zhou, Melkior Ornik, and Jun Liu. Resolvent-type data-driven learning of generators for unknown continuous-time dynamical systems. *arXiv preprint arXiv:2411.00923*, 2024.
- [19] Igor Mezić. Spectral Properties of Dynamical Systems, Model Reduction and Decompositions. *Nonlinear Dynamics*, 41(1-3):309–325, August 2005.
- [20] Amnon Pazy. *Semigroups of Linear Operators and Applications to Partial Differential Equations*, volume 44. Springer Science & Business Media, 2012.
- [21] William H Press. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.
- [22] Thanin Quartz, Ruikun Zhou, Hans De Sterck, and Jun Liu. Stochastic reinforcement learning with stability guarantees for control of unknown nonlinear systems. *arXiv preprint arXiv:2409.08382*, 2024.
- [23] Peter J Schmid. Dynamic mode decomposition of numerical and experimental data. *Journal of fluid mechanics*, 656:5–28, 2010.
- [24] Yoshihiko Susuki, Alexandre Mauroy, and Igor Mezic. Koopman resolvent: A laplace-domain analysis of nonlinear autonomous dynamical systems. *SIAM Journal on Applied Dynamical Systems*, 20(4):2013–2036, 2021.
- [25] Bhagyashree Umathe and Umesh Vaidya. Spectral koopman method for identifying stability boundary. *IEEE Control Systems Letters*, 2023.
- [26] Anthony Vannelli and Mathukumalli Vidyasagar. Maximal Lyapunov functions and domains of attraction for autonomous nonlinear systems. *Automatica*, 21(1):69–80, 1985.
- [27] 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*, 25:1307–1346, 2015.
- [28] Bowen Yi and Ian R Manchester. On the equivalence of contraction and koopman approaches for nonlinear stability and control. *IEEE Transactions on Automatic Control*, 69(7):4336–4351, 2023.
- [29] Ruikun Zhou, Maxwell Fitzsimmons, Yiming Meng, and Jun Liu. Physics-informed extreme learning machine Lyapunov functions. *IEEE Control Systems Letters*, 2024.
- [30] Ruikun Zhou, Thanin Quartz, Hans De Sterck, and Jun Liu. Neural lyapunov control of unknown nonlinear systems with stability guarantees. *Advances in Neural Information Processing Systems*, 35:29113–29125, 2022.
- [31] V. I. Zubov. *Methods of A. M. Lyapunov and Their Application*. Noordhoff, 1964.