

Numerical approximation of the Koopman-von Neumann equation: Operator learning and quantum computing

Stefan Klus¹, Feliks Nüske², and Patrick Gelß³

¹*School of Mathematical & Computer Sciences, Heriot-Watt University, Edinburgh, United Kingdom*

²*Max-Planck-Institute for Dynamics of Complex Technical Systems, Magdeburg, Germany*

³*Zuse Institute Berlin, Berlin, Germany*



(Received 9 April 2026; accepted 24 September 2026; published 13 October 2026)

The Koopman-von Neumann equation describes the evolution of wave functions associated with autonomous ordinary differential equations and can be regarded as a quantum physics-inspired formulation of classical mechanics. The main advantage compared to conventional transfer operators such as Koopman and Perron-Frobenius operators is that the Koopman-von Neumann operator is unitary even if the dynamics are non-Hamiltonian. Projecting this operator onto a finite-dimensional subspace allows us to represent it by a unitary matrix, which in turn can be expressed as a quantum circuit. We will exploit relationships between the Koopman-von Neumann framework and classical transfer operators to derive numerical methods to approximate the Koopman-von Neumann operator and its eigenvalues and eigenfunctions from data. Furthermore, we will show that the choice of basis functions and domain are crucial to ensure that the operator is well-defined. We will illustrate the results with the aid of guiding examples, including simple undamped and damped oscillators and the Lotka-Volterra model.

DOI: [10.1103/23fi-9hw8](https://doi.org/10.1103/23fi-9hw8)

I. INTRODUCTION

In recent years, the field of quantum computing has advanced rapidly. Exploiting quantum mechanical effects such as superposition and entanglement, quantum computers are able to solve certain types of problems faster than their classical counterparts. More specifically, quantum speedups have been rigorously established for a limited number of structured problem classes, most notably integer factorization, discrete logarithms, and related order-finding tasks [1,2], whereas unstructured search and several Monte Carlo-type estimation problems admit at most quadratic quantum improvements [3,4]. There is currently no evidence that quantum computers can efficiently solve general NP-hard problems. Moreover, many other proposed quantum advantages rely on additional assumptions on data access, sparsity, conditioning, or output representation [5,6]. A natural question thus is whether quantum computers can be used to speed up the simulation and analysis of classical nonlinear dynamical systems. One possible approach is to utilize the Koopman-von Neumann framework, a quantum physics-inspired formulation of classical mechanics [7–12]. While Koopman and von Neumann originally considered only Hamiltonian systems—in this case the Koopman operator itself is already unitary—the approach has been extended to arbitrary autonomous ordinary differential equations. Instead of analyzing the Koopman operator,

which propagates observables, or its adjoint, the Perron-Frobenius operator, which propagates probability densities, it is possible to derive a unitary operator, the Koopman-von Neumann operator,¹ that describes the evolution of complex-valued wave functions. The corresponding probability densities can then be obtained by applying Born's rule.

Projecting the Koopman-von Neumann operator onto a space spanned by a set of basis functions allows us to represent it by a unitary matrix, which in turn can be encoded by a quantum circuit. It is thus possible to simulate the action of the Koopman-von Neumann operator on wave functions using quantum computing. However, there are important differences between the behavior of the Koopman-von Neumann equation and the standard Schrödinger equation as pointed out in Ref. [10]. Most importantly, the Koopman-von Neumann generator is just a first-order differential operator unlike the Schrödinger operator. Furthermore, neither Planck's constant $\hbar$ nor a phase is involved in this fully classical theory [12]. We will analyze the feasibility of Koopman-von Neumann mechanics for describing the evolution of deterministic dynamical systems—including non-Hamiltonian dynamics—and explore relationships with the closely related Koopman and Perron-Frobenius operators and their infinitesimal generators.

While conventional transfer operators such as the Koopman operator and Perron-Frobenius operator are in general

Published by the American Physical Society under the terms of the Creative Commons Attribution 4.0 International license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

¹We will call the propagator for a fixed lag time *Koopman-von Neumann operator* and the associated infinitesimal generator of the semigroup of operators *Koopman-von Neumann generator* so that this is consistent with the notation used for the other operators and their generators.

well understood and have been used extensively (and successfully) in the past, see Refs. [13–21] to name just a few, the Koopman-von Neumann framework remains largely unexplored and almost no real-world applications can be found in the literature. Rigorous functional-analytic studies of the Koopman-von Neumann equation have emerged only recently. In particular, the well-posedness of the associated initial value problem for dynamical systems on bounded domains has been investigated in Ref. [22], clarifying the relationships with classical transport equations and transfer operator theory. Further works study the Koopman-von Neumann framework from an operator-theoretic perspective and consider structural properties of the corresponding Liouville operators in specific dynamical settings [23,24]. Another active line of research investigates the Koopman-von Neumann formalism in an algorithmic setting, addressing implementations on quantum hardware and in the form of circuit-based simulations [10,25–27]. These results establish important theoretical foundations and provide a starting point for the development and analysis of numerical approximation schemes. Classical numerical simulation methods and data-driven techniques for the analysis of the Koopman-von Neumann equation have received little attention in the literature so far.

We are in particular interested in spectral properties since the eigenvalues and eigenfunctions of conventional transfer operators contain important information about dominant timescales or slowly evolving spatiotemporal modes, but also in the propagation of probability densities, observables, and wave functions. Although the Koopman operator has been known for a long time, its recent rise in popularity can be explained by the availability of efficient machine learning algorithms for estimating transfer operators from data and the abundance of training data. This allows us to gain insights into global properties of the underlying system without requiring detailed mathematical models. One of the most frequently used methods for approximating transfer operators is *extended dynamic mode decomposition* [18,19], which can be regarded as a nonlinear variant of *dynamic mode decomposition* [28,29]. There are various kernel-based extensions [30,31], deep learning formulations [32,33], and methods to directly approximate infinitesimal generators of transfer operators [34,35]. Closely related methods have been developed independently by the signal processing, climate science, and molecular dynamics communities, see Refs. [36–38]. Comprehensive reviews of DMD-type algorithms can be found in Refs. [39,40]. Our goal is to show how variants of these methods can also be used to approximate the Koopman-von Neumann operator or its generator. The main contributions of this work are as follows:

(1) We analyze relationships between infinitesimal generators of conventional transfer operators and the Koopman-von Neumann framework and derive Galerkin projections of these operators.

(2) We approximate the Koopman-von Neumann generator using data-driven methods as well as finite element methods, which enables us not only to compute eigenvalues and eigenfunctions, but also to propagate wave functions and thus probability densities.

(3) To illustrate the results and to highlight the advantages and limitations of the proposed methods, we consider simple guiding examples and benchmark problems, including undamped and damped oscillators as well as the Lotka-Volterra model.

(4) We show that choosing suitable basis functions for the undamped oscillator allows us to construct a quantum circuit representation of the propagator of the projected Koopman-von Neumann generator.

It is well known that Galerkin projections of transfer operators will in general violate conservation laws or cause numerical artifacts such as spurious oscillations, unless finite-dimensional invariant subspaces are known. We will show that at least for simple linear systems with conserved quantities, such invariant subspaces can be constructed for appropriately defined domains. It is then possible to analytically compute eigenvalues and eigenfunctions and to correctly predict the evolution of probability densities, observables, and wave functions contained in the invariant subspace. For more complicated systems, such subspaces can in general not be easily constructed. Choosing suitable projections that preserve the dynamical properties of interest and faithfully simulating such systems on quantum computers remains an open problem [12].

While we will focus specifically on deterministic dynamical systems given by autonomous ordinary differential equations, nondeterministic processes governed by stochastic differential equations would be of immense interest as well. Under certain conditions, the Koopman generator associated with a stochastic differential equation can be transformed into a Schrödinger operator and vice versa. This transformation was used in Refs. [41,42] to apply data-driven methods developed for the approximation of the Koopman operator or generator to quantum systems. In the same way, we could represent a stochastic process by a Schrödinger equation and apply numerical methods developed for quantum systems. Analyzing relationships between these transformations and the Koopman-von Neumann framework, however, is beyond the scope of this work.

We will introduce the Perron-Frobenius, Koopman, and Koopman-von Neumann generators in Sec. II. Furthermore, we will highlight similarities but also major differences between these operators. In Sec. III, we will derive Galerkin projections and in particular data-driven approximations of the Koopman-von Neumann generator. Linear as well as nonlinear benchmark problems will be discussed in Sec. IV. Numerical results—using either data-driven approximations or finite element methods—will be presented in Sec. V. We will conclude with open questions and future work in Sec. VI.

II. TRANSFER OPERATORS AND KOOPMAN-VON NEUMANN MECHANICS

In this section, we will introduce the Koopman-von Neumann framework, starting from the Koopman operator description of classical systems. The idea behind Koopman-von Neumann mechanics is to provide quantum representations of classical systems, such that the statistics of the classical dynamics can be recovered by applying the rules of quantum

TABLE I. Operators and their eigenvalues and eigenfunctions.

	Differential operator	Eigenvalue	Eigenfunction
Koopman generator	$\mathcal{L}f = b \cdot \nabla f$	λ	f_λ
Perron-Frobenius generator	$\mathcal{L}^*\rho = -b \cdot \nabla \rho - \text{div}(b)\rho$	μ	ρ_μ
Koopman-von Neumann generator	$\mathcal{Q}\psi = -b \cdot \nabla \psi - \frac{1}{2}\text{div}(b)\psi$	ν	ψ_ν

mechanics. In what follows, we will use the notation summarized in Table I.

A. Koopman and Perron-Frobenius generators

We begin by providing a high-level description of the Koopman operator framework for general autonomous ordinary differential equations.

1. Autonomous ordinary differential equations

Let $\Omega \subseteq \mathbb{R}^d$ be the state space. Given an autonomous ordinary differential equation of the form

$$\dot{x} = b(x), \quad (1)$$

with $b : \Omega \rightarrow \mathbb{R}^d$ sufficiently smooth, and an initial condition $x(0) = x_0$, the semigroup of Koopman operators $\{\mathcal{K}^t\}$ is given by

$$(\mathcal{K}^t f)(x) = f(\Phi^t(x)),$$

where Φ^t is the flow map defined by $\Phi^t(x(0)) = x(t)$. The infinitesimal generator $\mathcal{L}$ can then be written as

$$\mathcal{L}f = b \cdot \nabla f = \sum_{i=1}^d b_i \frac{\partial f}{\partial x_i} \quad (2)$$

and its adjoint $\mathcal{L}^*$ —the generator of the semigroup of Perron-Frobenius operators—as

$$\mathcal{L}^*\rho = -\text{div}(b\rho) = -\sum_{i=1}^d \frac{\partial(b_i\rho)}{\partial x_i}. \quad (3)$$

Using the product rule, we can also write $\mathcal{L}^*\rho = -b \cdot \nabla \rho - \text{div}(b)\rho$. The operator $\mathcal{L}$ describes the evolution of observables and $\mathcal{L}^*$ the evolution of probability densities. For a detailed introduction, see Refs. [13,35].

2. Hamiltonian systems

As an important illustration, consider a Hamiltonian system defined by a classical Hamiltonian $H : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$, governed by Hamilton's equation of motion

$$\dot{q}_i = \frac{\partial H}{\partial p_i} \quad \text{and} \quad \dot{p}_i = -\frac{\partial H}{\partial q_i},$$

with $i = 1, \dots, d$. The vector $q \in \mathbb{R}^d$ contains the generalized coordinates and $p \in \mathbb{R}^d$ the corresponding momenta. The Hamiltonian formalism is a convenient mathematical tool to describe conservative mechanical systems [43]. Such systems—and thus also the associated transfer operators—have many special properties.

Defining $x = \begin{bmatrix} q \\ p \end{bmatrix} \in \mathbb{R}^{2d}$, we obtain $b = \begin{bmatrix} \nabla_p H \\ -\nabla_q H \end{bmatrix}$ and the Koopman generator can be written as

$$\mathcal{L}f = \begin{bmatrix} \nabla_p H \\ -\nabla_q H \end{bmatrix} \cdot \begin{bmatrix} \nabla_q f \\ \nabla_p f \end{bmatrix} = \sum_{i=1}^d \left(\frac{\partial H}{\partial p_i} \frac{\partial f}{\partial q_i} - \frac{\partial H}{\partial q_i} \frac{\partial f}{\partial p_i} \right).$$

Similarly, for the Perron-Frobenius generator, we obtain

$$\mathcal{L}^*\rho = -\text{div} \left(\begin{bmatrix} \nabla_p H \\ -\nabla_q H \end{bmatrix} \rho \right) = -\sum_{i=1}^d \left(\frac{\partial H}{\partial p_i} \frac{\partial \rho}{\partial q_i} - \frac{\partial H}{\partial q_i} \frac{\partial \rho}{\partial p_i} \right).$$

Note that the second-order derivatives $\frac{\partial^2 H}{\partial p_i \partial q_i}$ cancel out. It holds that $\mathcal{L}H = 0$ and $\mathcal{L} \exp(-H) = 0$. Eigenfunctions of $\mathcal{L}$ associated with the eigenvalue $\lambda = 0$ represent conservation laws; see Ref. [44] for more details. The crucial observation is that in this case $\mathcal{L}^* = -\mathcal{L}$, i.e., $\mathcal{L}$ is a skew-adjoint operator [13]. It then follows that $\mathcal{H} = i\mathcal{L}$ is a Hermitian operator, which can be used to define a quantum system mirroring the dynamical properties of the classical dynamics. We will show in the next section how to generalize this idea beyond Hamiltonian systems.

B. Koopman-von Neumann mechanics

Instead of considering the evolution of the probability density ρ , Koopman-von Neumann mechanics describes the system's state by a wave function ψ , from which we can compute the probability density ρ using Born's rule, i.e., $\rho = \psi^* \psi$. To make the introduction of the Koopman-von Neumann framework self-contained, we will include a short derivation. A detailed comparison of the different operators and their properties can be found in Ref. [12].

1. Autonomous ordinary differential equations

Although Koopman and von Neumann initially focused on Hamiltonian systems, it is possible to derive an evolution equation $\dot{\psi} = \mathcal{Q}\psi$, the so-called *Koopman-von Neumann equation* (the letter $\mathcal{Q}$ stands for quantum here), for general ordinary differential equations of the form (1), where

$$\mathcal{Q}\psi = -b \cdot \nabla \psi - \frac{1}{2}\text{div}(b)\psi = -\sum_{i=1}^d \left(b_i \frac{\partial \psi}{\partial x_i} + \frac{1}{2} \frac{\partial b_i}{\partial x_i} \psi \right), \quad (4)$$

see, e.g., Refs. [10,12]. Then $\rho := \psi^* \psi$ satisfies the Liouville equation $\dot{\rho} = \mathcal{L}^*\rho$. This can be shown using the product

rule:

$$\begin{aligned}
 \dot{\rho} &= - \sum_{i=1}^d \left(b_i \frac{\partial \psi^*}{\partial x_i} + \frac{1}{2} \frac{\partial b_i}{\partial x_i} \psi^* \right) \psi \\
 &\quad - \psi^* \sum_{i=1}^d \left(b_i \frac{\partial \psi}{\partial x_i} + \frac{1}{2} \frac{\partial b_i}{\partial x_i} \psi \right) \\
 &= - \sum_{i=1}^d \left(b_i \left[\frac{\partial \psi^*}{\partial x_i} \psi + \psi^* \frac{\partial \psi}{\partial x_i} \right] + \frac{\partial b_i}{\partial x_i} \psi^* \psi \right) \\
 &= - \sum_{i=1}^d \left(b_i \frac{\partial \rho}{\partial x_i} + \frac{\partial b_i}{\partial x_i} \rho \right) \\
 &= -b \cdot \nabla \rho - \text{div}(b) \rho = \mathcal{L}^* \rho.
 \end{aligned}$$

Multiplying the Koopman-von Neumann equation by the imaginary unit i results in

$$i\dot{\psi} = i\mathcal{Q}\psi := \mathcal{H}\psi$$

so that the equation closely resembles the time-dependent Schrödinger equation, i.e., $\mathcal{H}$ is a Hermitian operator and the corresponding propagator is unitary [12].

Remark II.1. Defining $\mathcal{L}_s = \frac{1}{2}(\mathcal{L} + \mathcal{L}^*)$ and $\mathcal{L}_a = \frac{1}{2}(\mathcal{L} - \mathcal{L}^*)$ to be the symmetric and skew-symmetric parts of the operator $\mathcal{L}$, we have $\mathcal{L} = \mathcal{L}_s + \mathcal{L}_a$ and $\mathcal{L}^* = \mathcal{L}_s - \mathcal{L}_a$. It then follows that

$$\mathcal{Q}\psi = -\mathcal{L}_a\psi = \frac{1}{2}(\mathcal{L}^* - \mathcal{L})\psi = -b \cdot \nabla \psi - \frac{1}{2}\text{div}(b)\psi,$$

i.e., the Koopman-von Neumann generator is the adjoint of the skew-symmetric part of the Koopman generator. This decomposition, along with a so-called *warped-phase transformation*, which introduces an additional one-dimensional variable, was also used in the Schrödingerization approach proposed in Ref. [45] to solve arbitrary linear partial differential equations on quantum computers. We only require the skew-symmetric part in what follows. A complete Koopman-based quantization procedure, named quantum embedding of classical dynamics (QECD), for ergodic measure-preserving systems was introduced in Ref. [46]. As we will show in Appendix A, the Koopman-von Neumann approach described here can be seen as an extension of QECD to general autonomous ordinary differential equations.

2. Hamiltonian systems

Since the divergence of a Hamiltonian vector field is zero, we immediately obtain

$$\mathcal{Q} = \mathcal{L}^* = -\mathcal{L}.$$

That is, the Koopman generator is in this case already skew-adjoint and the corresponding propagator automatically unitary.

C. Properties of transfer operators

The spectral properties of the Koopman operator are well-understood, see, e.g., Ref. [16]. We include short proofs of the corresponding properties of its infinitesimal generator for the sake of completeness. The question now is how eigenvalues

and eigenfunctions of the operators introduced above are related and whether we can apply numerical methods developed for the Koopman generator also to the Koopman-von Neumann generator.

Lemma II.1. Assuming that the functions constructed below are well-defined, the operators have the following properties:

(1) Product rule for the Koopman generator: It holds that $\mathcal{L}(fg) = (\mathcal{L}f)g + f(\mathcal{L}g)$.

(2) Products of eigenfunctions are eigenfunctions: Given two eigenfunctions f_{λ_1} and f_{λ_2} of $\mathcal{L}$ associated with the eigenvalues λ_1 and λ_2 , respectively, it holds that $\mathcal{L}(f_{\lambda_1}f_{\lambda_2}) = (\lambda_1 + \lambda_2)f_{\lambda_1}f_{\lambda_2}$.

(3) Smooth transformations retain conservation laws: Assume that $\mathcal{L}f_\lambda = \lambda f_\lambda$ and let $g: \mathbb{R} \rightarrow \mathbb{R}$ be differentiable, then $\mathcal{L}(g \circ f_\lambda) = \lambda f_\lambda \cdot g' \circ f_\lambda$. In particular, if $\lambda = 0$, then $\mathcal{L}(g \circ f_\lambda) = 0$.

(4) Powers of eigenfunctions are eigenfunctions: For $r \in \mathbb{R}$ we have f_λ^r is an eigenfunction associated with the eigenvalue $r\lambda$.

(5) Conservation laws generate additional eigenfunctions of the Perron-Frobenius generator: Let $\mathcal{L}f_0 = 0$ and $\mathcal{L}^*\rho_\mu = \mu\rho_\mu$, then $\mathcal{L}^*(f_0\rho_\mu) = \mu(f_0\rho_\mu)$.

(6) Generalized Born rule for the Koopman-von Neumann generator: We have $\mathcal{L}^*(fg) = (\mathcal{Q}f)g + f(\mathcal{Q}g)$.

(7) Born rule for eigenfunctions: Given two eigenfunctions ψ_{v_1} and ψ_{v_2} of $\mathcal{Q}$ corresponding to the eigenvalues v_1 and v_2 , we have $\mathcal{L}^*(\psi_{v_1}\psi_{v_2}) = (v_1 + v_2)\psi_{v_1}\psi_{v_2}$.

(8) Eigenfunctions of systems with constant divergence: Assume $\text{div}(b) \equiv \beta$, where $\beta \in \mathbb{R}$ is a constant. Given $\mathcal{L}f_\lambda = \lambda f_\lambda$, we obtain $\mathcal{L}^*f_\lambda = -(\lambda + \beta)f_\lambda$ and $\mathcal{Q}f_\lambda = -(\lambda + \frac{1}{2}\beta)f_\lambda$.

The proofs of these properties can be found in Appendix B. Examples of systems with constant divergence are Hamiltonian systems, where $\beta = 0$, and linear differential equations of the form $\dot{x} = Bx$, where $\beta = \text{tr}(B)$. Linear differential equations—including linear Hamiltonian systems—will be discussed in more detail below.

D. Transfer operators on bounded domains

So far, we only formally introduced the above operators, but did not specify the domain Ω . Koopman and Perron-Frobenius operators are often defined on $L^2(\mathbb{R}^d)$, with the requirement that the observables or probability densities decay to zero for $\|x\| \rightarrow \infty$. In practice, however, we typically have to consider bounded domains Ω . If the domain is bounded, we assume Ω to be forward-invariant under the flow in what follows, i.e., $\Phi^t(\Omega) \subseteq \Omega$ for all $t \geq 0$, so that no boundary conditions are required for the Koopman generator [47]. A key question then is under which conditions the Perron-Frobenius generator retains the same analytical expression as in Eq. (3). The following result shows that imposing homogeneous Dirichlet boundary conditions on probability densities is sufficient. This condition is natural given the forward-invariance of the dynamics.

Lemma II.2. Given a bounded domain Ω , let ρ satisfy homogeneous Dirichlet boundary conditions, i.e., $\rho(x) = 0$ on $\partial\Omega$, then $\langle \rho, \mathcal{L}f \rangle = \langle \mathcal{L}^*\rho, f \rangle$, where $\mathcal{L}^*\rho = -\text{div}(b\rho)$.

Proof. We follow the proof detailed in Ref. [13], with the only difference that we consider a bounded domain Ω . It holds that

$$\begin{aligned} \langle \rho, \mathcal{L}f \rangle &= \int_{\Omega} \rho [b \cdot \nabla f] dx \\ &= \int_{\Omega} \operatorname{div}(b \rho f) dx - \int_{\Omega} \operatorname{div}(b \rho) f dx = \langle \mathcal{L}^* \rho, f \rangle, \end{aligned}$$

since, using the divergence theorem and the fact that ρ is by definition zero on the boundary,

$$\int_{\Omega} \operatorname{div}(b \rho f) dx = \int_{\partial \Omega} (b \rho f) \cdot \vec{n} ds = 0,$$

where $\vec{n}$ is the outward-pointing normal vector. ■

This result is important since it allows us to define the Koopman-von Neumann generator $\mathcal{Q}$ in terms of the Koopman and Perron-Frobenius generators (see Remark II.1), now restricted to the space of functions satisfying homogeneous Dirichlet boundary conditions. The formal definition (4) is valid for any smooth function satisfying zero boundary conditions. A complete characterization of the domain of definition for the Koopman-von Neumann generator on bounded domains can be found in Ref. [22], where Perron-Frobenius-Sobolev (PFS) spaces are introduced as the natural function space setting for the bounded-domain case. Rather than working only with classical Sobolev spaces, the analysis uses PFS spaces along with an appropriate vanishing-trace condition to incorporate the boundary behavior in a way that is consistent with the Perron-Frobenius framework, with the no-outflow condition providing the boundary-based analog of forward invariance by ensuring that trajectories do not leave the domain under the forward flow.

E. Linear dynamical systems and invariant subspaces

To study the Koopman-von Neumann generator and its properties in more depth, we first consider linear dynamical systems since a wealth of their properties can be computed explicitly, see, e.g., Refs. [16,19].

Lemma II.3. Given $\dot{x} = Bx$, where $B \in \mathbb{R}^{d \times d}$, let $w \in \mathbb{C}^d$ be an eigenvector of B^T corresponding to the eigenvalue $\lambda \in \mathbb{C}$, i.e., $B^T w = \lambda w$. Then $f_{\lambda}(x) = x \cdot w$ is an eigenfunction of the Koopman generator.²

Proof. We have $\mathcal{L}f_{\lambda}(x) = Bx \cdot w = x \cdot B^T w = x \cdot (\lambda w) = \lambda f_{\lambda}(x)$. ■

In what follows, we will call the eigenfunctions given by the eigenvectors of B^T *principal eigenfunctions*. Note that these functions are not bounded if we consider $\Omega = \mathbb{R}^d$ and do in general not satisfy the boundary conditions if the domain Ω is bounded. We can now construct infinitely many additional eigenfunctions by considering products of these eigenfunctions as shown in Lemma II.1. As a special case, we will analyze Hamiltonian systems.

Definition II.1. Let $J \in \mathbb{R}^{2d \times 2d}$ be the matrix given by

$$J = \begin{bmatrix} 0 & I \\ -I & 0 \end{bmatrix},$$

then $B \in \mathbb{R}^{2d \times 2d}$ is called *Hamiltonian* if $JB + B^T J = 0$. Note that $J^{-1} = J^T = -J$.

Writing B as a block matrix

$$B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix},$$

it is Hamiltonian if $B_{11}^T + B_{22} = 0$ and B_{12} and B_{21} are symmetric [43]. It then follows that $\dot{x} = Bx$ is a Hamiltonian system with

$$H(q, p) = -\frac{1}{2} q^T B_{21} q + p^T B_{11} q + \frac{1}{2} p^T B_{12} p.$$

The spectrum of Hamiltonian matrices is symmetric about 0, i.e., $\lambda \in \sigma(B) \Rightarrow -\lambda \in \sigma(B)$. If all eigenvalues of B lie on the imaginary axis, then solutions are purely oscillatory.

Example II.1. A *damped oscillator* is given by $\ddot{q} + \gamma \dot{q} + \omega^2 q = 0$, where γ is the damping ratio and ω the angular frequency. Defining $x = \begin{bmatrix} q \\ p \end{bmatrix}$, this can be rewritten as

$$\dot{x} = \underbrace{\begin{bmatrix} 0 & 1 \\ -\omega^2 & -\gamma \end{bmatrix}}_{=:B} x.$$

We obtain the *undamped oscillator* by setting $\gamma = 0$. The Hamiltonian of the system is in this case given by $H(q, p) = \frac{1}{2} \omega^2 q^2 + \frac{1}{2} p^2$.

We will now construct invariant subspaces for linear systems, which enable us to define suitable dictionaries for the numerical approximation techniques derived in Sec. III. The first result shows that finite-dimensional subspaces spanned by monomials up to a fixed order r are invariant under the action of the Koopman generator.

Lemma II.4. Let $\alpha = (\alpha_1, \dots, \alpha_d) \in \mathbb{N}_0^d$ be a multi-index with $|\alpha| = \sum_{i=1}^d \alpha_i$ and define

$$\mathbb{V} = \operatorname{span}\{x^{\alpha} = x_1^{\alpha_1} \cdots x_d^{\alpha_d} : |\alpha| \leq r\}$$

to be the finite-dimensional subspace spanned by monomials of order up to r . Given a dynamical system of the form $\dot{x} = Bx$ and a function $f \in \mathbb{V}$, it holds that $\mathcal{L}f \in \mathbb{V}$.

Proof. Let $e_i \in \mathbb{N}_0^d$ denote the i th unit vector. First, assume that $f(x) = x^{\alpha}$, with $|\alpha| = q \leq r$, then

$$\frac{\partial f}{\partial x_i} = \begin{cases} \alpha_i x^{\alpha - e_i}, & \alpha_i \neq 0, \\ 0, & \text{otherwise,} \end{cases}$$

is either a monomial of order $q - 1$ or identical to zero. Thus,

$$\mathcal{L}f = Bx \cdot \nabla f = \sum_{i=1}^d \sum_{j=1}^d B_{ij} \frac{\partial f}{\partial x_i} x_j$$

is at most a polynomial of degree $q \leq r$ and thus contained in $\mathbb{V}$. For an arbitrary function $f \in \mathbb{V}$, we apply the same logic to each monomial term. ■

We can now use this results to construct basis functions that automatically satisfy the boundary conditions if the domain is bounded.

Lemma II.5. Assume that there exists a conservation law f_0 , i.e., $\mathcal{L}f_0 = 0$, that vanishes on the boundary, that is, $f_0(x) = 0$ for all $x \in \partial\Omega$. We define a new function space

$$\mathbb{V}' = \operatorname{span}\{f_0 f : f \in \mathbb{V}\}$$

²We follow the physics convention and define the inner product to be conjugate-linear in the first variable.

so that all functions satisfy the boundary conditions. Given $f' \in \mathbb{V}'$, it holds that $\mathcal{L}f' \in \mathbb{V}'$.

Proof. Using Lemma II.1 and Lemma II.4, we have

$$\mathcal{L}(f_0 f) = \underbrace{(\mathcal{L}f_0)f}_{=0} + f_0 \underbrace{(\mathcal{L}f)}_{\in \mathbb{V}} \in \mathbb{V}'.$$

The same result also applies to the Perron-Frobenius and Koopman-von Neumann generators as the following corollary shows.

Corollary II.1. The function space $\mathbb{V}'$ is also invariant under the action of the Perron-Frobenius generator $\mathcal{L}^*$ and Koopman-von Neumann generator $\mathcal{Q}$.

Proof. Let $\psi \in \mathbb{V}'$, then

$$\mathcal{Q}\psi = -\underbrace{b \cdot \nabla \psi}_{=\mathcal{L}\psi \in \mathbb{V}'} - \frac{1}{2} \underbrace{\text{div}(b)\psi}_{=\text{tr}(B)\psi \in \mathbb{V}'} \in \mathbb{V}'.$$

The proof for $\mathcal{L}^*$ follows in the same way. ■

Given a conservation law f_0 , this allows us to define a suitable domain by choosing the boundary $\partial\Omega$ to be a level set of the function f_0 , provided it generates a valid bounded domain. Invariant subspaces for the operators $\mathcal{L}$, $\mathcal{L}^*$, and $\mathcal{Q}$ can then be constructed as we will illustrate in Sec. IV.

III. APPROXIMATION OF THE KOOPMAN-VON NEUMANN GENERATOR

Numerical methods for the approximation of the Koopman operator and its infinitesimal generator gained a lot of attention in recent years, see, e.g., Refs. [18,35]. We can now exploit the relationships between the operators derived in Sec. II to estimate the Koopman-von Neumann generator.

A. Galerkin projection

Let $\{\phi_i\}_{i=1}^n$ be a set of linearly independent basis functions that, if the domain is bounded, satisfy the homogeneous Dirichlet boundary conditions. Define $\mathbb{V} = \text{span}\{\phi_i\}_{i=1}^n$ to be the generated n -dimensional subspace and

$$\phi(x) = [\phi_1(x), \dots, \phi_n(x)]^\top \in \mathbb{R}^n.$$

We can write any function $f \in \mathbb{V}$ as a linear combination of the basis functions, i.e.,

$$f(x) = \sum_{i=1}^n c_i \phi_i(x) = c^\top \phi(x),$$

where $c = [c_1, \dots, c_n]^\top \in \mathbb{R}^n$. Let $\mathcal{P}$ denote the projection onto $\mathbb{V}$, then the matrix representation $L \in \mathbb{R}^{n \times n}$ of the Galerkin approximation $\mathcal{L}|_{\mathbb{V}} := \mathcal{P}\mathcal{L}\mathcal{P}$ of the Koopman generator $\mathcal{L}$ is given by

$$L = G^{-1}A,$$

where

$$G_{ij} = \langle \phi_i, \phi_j \rangle \quad \text{and} \quad A_{ij} = \langle \phi_i, \mathcal{L}\phi_j \rangle.$$

We can also write this in compact form as

$$G = \int \phi(x)\phi(x)^\top dx \quad \text{and} \quad A = \int \phi(x)\mathcal{L}\phi(x)^\top dx,$$

where $\mathcal{L}$ is applied component-wise. For a function $f(x) = c^\top \phi(x)$, we then obtain

$$\mathcal{L}|_{\mathbb{V}} f(x) = (Lc)^\top \phi(x).$$

Since $\langle \phi_i, \mathcal{L}^* \phi_j \rangle = \langle \mathcal{L}\phi_i, \phi_j \rangle = A_{ji}$, the matrix representation L^* of the projected Perron-Frobenius generator $\mathcal{L}^*|_{\mathbb{V}}$ is given by

$$L^* = G^{-1}A^\top.$$

Similarly, to approximate the projected Koopman-von Neumann generator $\mathcal{Q}|_{\mathbb{V}}$, we can compute $\langle \phi_i, \mathcal{Q}\phi_j \rangle = \frac{1}{2}(\langle \phi_i, \mathcal{L}^* \phi_j \rangle - \langle \phi_i, \mathcal{L}\phi_j \rangle) = \frac{1}{2}(A_{ji} - A_{ij})$ so that its matrix representation is given by

$$Q = \frac{1}{2}G^{-1}(A^\top - A).$$

The matrix Q is now skew-symmetric with respect to the G -weighted inner product $\langle \alpha, \beta \rangle_G = \alpha^\top G \beta$ but not the standard Euclidean inner product $\langle \alpha, \beta \rangle = \alpha^\top \beta$. We can enforce skew-symmetry with respect to the unweighted inner product by applying a so-called whitening transformation. Since the basis functions are by assumption linearly independent, the matrix G is symmetric and positive definite so that its eigenvalues are positive. Let $G = V D V^\top$ be the eigendecomposition of G and define $\tilde{\phi}(x) = D^{-\frac{1}{2}} V^\top \phi(x)$, then

$$\begin{aligned} \tilde{G} &= \int \tilde{\phi}(x) \tilde{\phi}(x)^\top dx = \int D^{-\frac{1}{2}} V^\top \phi(x) \phi(x)^\top V D^{-\frac{1}{2}} dx \\ &= D^{-\frac{1}{2}} V^\top G V D^{-\frac{1}{2}} = I \end{aligned}$$

and

$$\begin{aligned} \tilde{A} &= \int \tilde{\phi}(x) \mathcal{L} \tilde{\phi}(x)^\top dx = \int D^{-\frac{1}{2}} V^\top \phi(x) \mathcal{L} \phi(x)^\top V D^{-\frac{1}{2}} dx \\ &= D^{-\frac{1}{2}} V^\top A V D^{-\frac{1}{2}}, \end{aligned}$$

i.e., the transformed basis functions $\tilde{\phi}_i(x)$ are orthonormal and $\tilde{Q} = \frac{1}{2} \tilde{G}^{-1}(\tilde{A}^\top - \tilde{A}) = \frac{1}{2}(\tilde{A}^\top - \tilde{A})$. This shows that it is always possible to define the basis functions in such a way that the matrix representation of the projected Koopman-von Neumann generator is skew-symmetric with respect to the standard Euclidean inner product so that the corresponding propagator $e^{\tilde{Q}}$ is unitary.

To compute eigenvalues and eigenfunctions of the projected Koopman-von Neumann generator $\mathcal{Q}|_{\mathbb{V}}$, we have to compute eigenvalues and eigenvectors of Q . Given $Qv = v$, this implies that $\psi_v(x) = v^\top \phi(x)$ is an eigenfunction since

$$\mathcal{Q}|_{\mathbb{V}} \psi_v(x) = (Qv)^\top \phi(x) = v^\top \phi(x) = v \psi_v(x).$$

Additionally, the temporal evolution of the coefficients c of a function $\psi(x) = c^\top \phi(x)$ can now be described by the system of linear ordinary differential equations $\dot{c} = Qc$. This allows us to propagate projected wave functions in time.

B. Data-driven approximation

A data-driven approach to approximate the Koopman generator called *generator extended dynamic mode decomposition* (gEDMD), which can be viewed as a Galerkin approximation, where the matrices A and G are estimated using Monte Carlo integration, was proposed in Ref. [35]. Our goal is to extend

this method to the Koopman-von Neumann generator. Given uniformly sampled training data $x^{(l)}$, with $l = 1, \dots, m$, and the corresponding time derivatives $\dot{x}^{(l)}$, which can, for instance, be estimated using finite difference approximations, we define

$$\dot{\phi}_k(x) = (\mathcal{L}\phi_k)(x) = \sum_{i=1}^d b_i(x) \frac{\partial \phi_k}{\partial x_i}(x)$$

and compute the matrices

$$\Phi_X = \begin{bmatrix} \phi_1(x_1) & \dots & \phi_1(x_m) \\ \vdots & \ddots & \vdots \\ \phi_n(x_1) & \dots & \phi_n(x_m) \end{bmatrix} \quad \text{and} \quad \dot{\Phi}_X = \begin{bmatrix} \dot{\phi}_1(x_1) & \dots & \dot{\phi}_1(x_m) \\ \vdots & \ddots & \vdots \\ \dot{\phi}_n(x_1) & \dots & \dot{\phi}_n(x_m) \end{bmatrix}.$$

The required derivatives of the basis functions can, for example, be computed using automatic differentiation. We then have

$$\begin{aligned} 4\hat{G} &:= \frac{1}{m} \Phi_X \Phi_X^\top \\ &= \frac{1}{m} \sum_{l=1}^m \phi(x^{(l)}) \phi(x^{(l)})^\top \xrightarrow{m \rightarrow \infty} \int \phi(x) \phi(x)^\top dx = G, \\ \hat{A} &:= \frac{1}{m} \Phi_X \dot{\Phi}_X^\top \\ &= \frac{1}{m} \sum_{l=1}^m \phi(x^{(l)}) \dot{\phi}(x^{(l)})^\top \xrightarrow{m \rightarrow \infty} \int \phi(x) \mathcal{L}\phi(x)^\top dx = A, \end{aligned}$$

so that $\hat{L} = \hat{G}^+ \hat{A}$ is a consistent approximation of the projected Koopman generator, where $^+$ denotes the pseudoinverse; see Ref. [35]. Similarly, we can compute empirical estimates of the projected Perron-Frobenius generator and Koopman-von Neumann generator using

$$\hat{L}^* = \hat{G}^+ \hat{A}^\top \quad \text{and} \quad \hat{Q} = \frac{1}{2} \hat{G}^+ (\hat{A}^\top - \hat{A}),$$

respectively. Using whitened basis functions as described above, we then obtain a skew-symmetric matrix representation of the Koopman-von Neumann generator computed from data.

Remark III.1. We would like to point out that:

(1) In the derivation above, we implicitly assumed that the data points $x^{(l)}$ are uniformly distributed. If we estimate the operators from trajectory data, then this is in general not the case and we compute a Galerkin projection with respect to a weighted inner product; see Refs. [18,19,48,49] for a more detailed analysis.

(2) A data-driven method for estimating the Koopman generator that does not require time-derivatives is described in Ref. [34]. The approximation of the generator is in this case obtained by taking the logarithm of a matrix representation of the Koopman operator estimated from time series data. The method could hence also be used to approximate the Koopman-von Neumann generator.

IV. BENCHMARK PROBLEMS

In this section, we will introduce and analyze benchmark problems, namely simple undamped and damped oscillators and the Lotka-Volterra model. We will in particular illustrate some of the similarities and structural differences between the Koopman and Koopman-von Neumann generators that occur for different systems and boundary conditions. Numerical results for these systems will be presented in Sec. V.

A. Undamped oscillator

We first consider the undamped oscillator defined in Example II.2 and choose $\omega = \sqrt{2}$ so that

$$B = \begin{bmatrix} 0 & 1 \\ -2 & 0 \end{bmatrix}.$$

Using Lemma II.3, the principal eigenvalues and eigenfunctions of the Koopman generator are

$$\begin{aligned} \lambda_1 &= i\sqrt{2}, & f_{\lambda_1}(x) &= ix_1 + \frac{1}{\sqrt{2}}x_2, \\ \lambda_2 &= -i\sqrt{2}, & f_{\lambda_2}(x) &= -ix_1 + \frac{1}{\sqrt{2}}x_2, \end{aligned}$$

with period $T = \frac{2\pi}{\sqrt{2}} \approx 4.44$. Additionally, any function of the form $f_\lambda(x) = f_{\lambda_1}(x)^{k_1} f_{\lambda_2}(x)^{k_2}$ is an eigenfunction corresponding to the eigenvalue $\lambda = k_1 \lambda_1 + k_2 \lambda_2 = i\sqrt{2}(k_1 - k_2)$. All these eigenvalues lie on the imaginary axis. Note that if $k_1 = k_2$, then $\lambda = 0$. In particular, for $k_1 = k_2 = 1$, we obtain $f_\lambda(x) = f_{\lambda_1}(x) f_{\lambda_2}(x) = x_1^2 + \frac{1}{2}x_2^2 = H(q, p) =: H(x)$, i.e., the Hamiltonian itself. We now show how to construct eigenfunctions satisfying boundary conditions in different settings:

(1) Let $\Omega = \mathbb{R}^2$, then the Gaussian-like function

$$f_0(x) = \sum_{k=0}^{\infty} \frac{(-H(x))^k}{k!} = \exp\left(-x_1^2 - \frac{1}{2}x_2^2\right)$$

is a conservation law, which respects zero boundary conditions at infinity. To construct additional eigenfunctions, we can multiply f_0 by powers of f_{λ_1} and f_{λ_2} . These functions are bounded and tend to zero for $\|x\| \rightarrow \infty$.

(2) As an example with a bounded domain, consider $\Omega = \{x \in \mathbb{R}^2 : x_1^2 + \frac{1}{2}x_2^2 < 1\}$, i.e., an ellipse with semiminor axis 1 and semimajor axis $\sqrt{2}$. Then

$$f_0(x) = 1 - x_1^2 - \frac{1}{2}x_2^2 \quad (5)$$

is a conservation law that vanishes on the boundary (i.e., Lemma II.5 applies) and can again be multiplied by powers of f_{λ_1} and f_{λ_2} to obtain additional eigenfunctions that satisfy the boundary conditions.

A few eigenfunctions of the Koopman generator are shown in Fig. 1. Since $\mathcal{L}^* = \mathcal{Q} = -\mathcal{L}$ here and the constructed eigenfunctions for the bounded domain case satisfy the boundary conditions, these functions are also eigenfunctions of the Perron-Frobenius generator and Koopman-von Neumann generator, associated with the eigenvalues $\mu = \nu = -\lambda$.

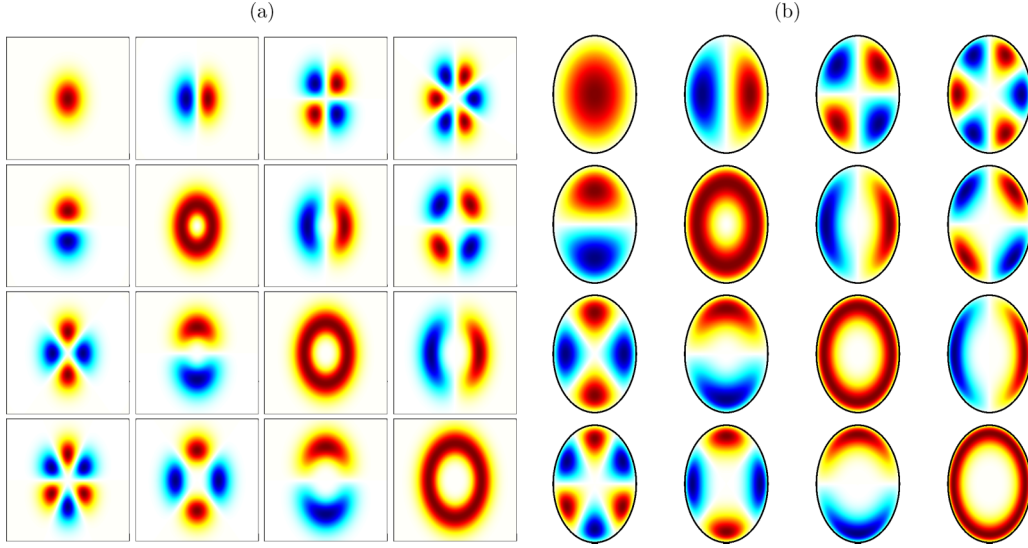


FIG. 1. Real part (if $k_1 \leq k_2$) or imaginary part (if $k_1 > k_2$ since the real part would be the same as that of the function with k_1 and k_2 swapped) of the eigenfunctions $f_\lambda(x) = f_0(x)f_{\lambda_1}(x)^{k_1}f_{\lambda_2}(x)^{k_2}$ of the Koopman generator (and hence also Koopman-von Neumann generator) associated with the undamped oscillator defined on the domains (a) $\Omega = \mathbb{R}^2$ and (b) $\Omega = \{x \in \mathbb{R}^2 : x_1^2 + \frac{1}{2}x_2^2 < 1\}$, where $k_1, k_2 \in \{0, 1, 2, 3\}$ are the row and column numbers, respectively. The eigenfunctions are plotted for (a) $x \in [-4, 4]^2$ and (b) $x \in \Omega$. Red represents positive and blue negative values. The corresponding eigenvalues are $\lambda = i\sqrt{2}(k_1 - k_2)$, i.e., multiples of the angular frequency ω . All eigenvalues are as expected purely imaginary.

B. Damped oscillator

Let us now consider the damped case. We again choose $\omega = \sqrt{2}$, but set $\gamma = 2$ so that

$$B = \begin{bmatrix} 0 & 1 \\ -2 & -2 \end{bmatrix}.$$

Using Lemma II.3, the principal eigenvalues and eigenfunctions of the Koopman generator are

$$\lambda_1 = -1 + i, \quad f_{\lambda_1}(x) = x_1 + \frac{1}{2}(1 - i)x_2,$$

$$\lambda_2 = -1 - i, \quad f_{\lambda_2}(x) = x_1 + \frac{1}{2}(1 + i)x_2.$$

By taking integer powers of these eigenfunctions, the grid of eigenvalues $\lambda = -(k_1 + k_2) + i(k_1 - k_2)$ and eigenfunctions shown in Fig. 2 can be generated. These eigenfunctions again grow to infinity with increasing $\|x\|$. Unlike for the undamped oscillator, we cannot easily find a conservation law in this case. We can, however, find a bounded domain $\Omega = \{x \in \mathbb{R}^2 : x_1^2 + x_1x_2 + \frac{1}{2}x_2^2 < 1\}$ that satisfies the forward-invariance condition. The boundary of this domain is a level set of the eigenfunction for $k_1 = k_2 = 1$, i.e., $f_\lambda = f_{\lambda_1}(x)f_{\lambda_2}(x) = x_1^2 + x_1x_2 + \frac{1}{2}x_2^2$, with eigenvalue $\lambda = -2$. The function f_λ is not a conservation law, and hence we cannot simply multiply the Koopman generator eigenfunctions by f_λ to obtain eigenfunctions of $\mathcal{L}^*$ or $\mathcal{Q}$. In fact, an eigenfunction of the Perron-Frobenius operator would be a Dirac distribution centered at the origin. However, such eigenfunctions capture only local information about the dynamics [47]. This example illustrates some of the structural differences of the spectra of $\mathcal{L}$ and $\mathcal{Q}$ that can occur.

Nevertheless, we can still use the Perron-Frobenius and Koopman-von Neumann operators or generators to propagate probability densities or wave functions. This will be illustrated in Sec. V.

Remark IV.1. It was shown in Ref. [47] that eigenfunctions of the Koopman operator or generator can be used to analyze the stability of dynamical systems. Indeed, the domain Ω chosen for the damped oscillator is forward-invariant and the origin $x^* = 0$ is a stable fixed point. A Lyapunov function is then an observable $V(x) > 0$ for $x \neq x^*$ that satisfies $\mathcal{L}V(x) < 0$ for all $x \neq x^*$. The eigenfunction for $k_1 = k_2 = 1$ with eigenvalue $\lambda = -2$ has exactly this property.

C. Lotka-Volterra model

As a prototypical nonlinear dynamical system, we choose the Lotka-Volterra model $\dot{x} = b(x)$, with

$$b(x) = \begin{bmatrix} x_1(1 - x_2) \\ x_2(x_1 - 1) \end{bmatrix}$$

and $\text{div}(b) = x_1 - x_2$, which describes the dynamics of two interacting species, one being a predator and the other its prey. Both the predator and prey populations oscillate around a fixed point. Unlike for the undamped oscillator, the period length now depends on an associated “energy” [50]. The Lotka-Volterra model can be turned into a Hamiltonian system by a nonlinear coordinate transformation, but in its original form the dynamics are not Hamiltonian. A well-known conserved quantity of the system is

$$f_0(x) = x_1 + x_2 - \log(x_1) - \log(x_2). \quad (6)$$

We define the domain $\Omega = \{x \in \mathbb{R}^2 : x_1 + x_2 - \log(x_1) - \log(x_2) < 3\}$ by a level set of f_0 , or, equivalently, a closed orbit of the system. It was shown in Ref. [51] that $\rho_0(x) = \frac{1}{x_1x_2}$ is an eigenfunction of the Perron-Frobenius generator with eigenvalue $\mu = 0$ and thus, using Lemma II.2, also the function

$$\tilde{\rho}_0(x) = (3 - f_0(x))\rho_0(x),$$

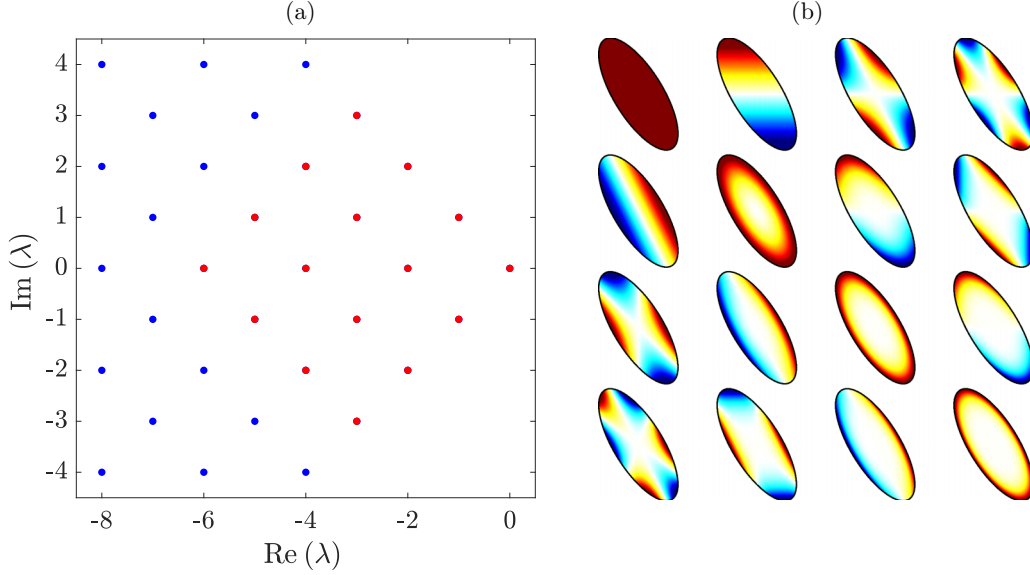


FIG. 2. (a) Lattice structure generated by the principal eigenvalues of the Koopman generator associated with the damped oscillator. The eigenvalues marked in red correspond to the eigenfunctions shown on the right. (b) Real or imaginary parts of the eigenfunctions $f_\lambda(x) = f_{\lambda_1}(x)^{k_1} f_{\lambda_2}(x)^{k_2}$ of the Koopman generator defined on $\Omega = \{x \in \mathbb{R}^2 : x_1^2 + x_1 x_2 + \frac{1}{2} x_2^2 < 1\}$. Note that the eigenfunctions do not vanish on the boundary. The corresponding eigenvalues are $\lambda = -(k_1 + k_2) + i(k_1 - k_2)$, where $k_1, k_2 \in \{0, 1, 2, 3\}$ are the row and column numbers, respectively.

which now satisfies the boundary conditions. This (unnormalized) density is invariant under the dynamics, which implies that its square-root is an invariant wave function.

V. NUMERICAL RESULTS

We will now approximate the Koopman-von Neumann generator using data-driven approaches as well as finite element methods. We then use the discretized generator to compute eigenvalues and eigenfunctions or to propagate wave functions.

A. Undamped oscillator

Let us first consider the undamped oscillator on the bounded domain Ω defined in Sec. IV. For this example, we will construct both an analytical discretization using a polynomial basis set and a generic approximation using random features, and then study their approximation properties. In addition, we use the analytical discretization to construct a quantum circuit representation for the Koopman-von Neumann operator $e^{t\hat{Q}}$.

1. Monomial basis

First, based on Lemma II.5, we select the basis functions

$$\phi(x) = \left(1 - x_1^2 - \frac{1}{2}x_2^2\right) \begin{bmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \\ x_1 x_2 \\ x_2^2 \end{bmatrix},$$

that is, polynomial functions up to degree two multiplied by the conservation law f_0 from Eq. (5). These basis functions exactly satisfy the Dirichlet boundary conditions on the domain and, as we have seen, they also define an invariant subspace for the Koopman-von Neumann generator. The Galerkin matrices G and A can in this case be computed analytically. We have

$$G = \sqrt{2}\pi \begin{bmatrix} \frac{1}{3} & 0 & 0 & \frac{1}{24} & 0 & \frac{1}{12} \\ 0 & \frac{1}{24} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{12} & 0 & 0 & 0 \\ \frac{1}{24} & 0 & 0 & \frac{1}{80} & 0 & \frac{1}{120} \\ 0 & 0 & 0 & 0 & \frac{1}{120} & 0 \\ \frac{1}{12} & 0 & 0 & \frac{1}{120} & 0 & \frac{1}{20} \end{bmatrix} \quad \text{and}$$

$$A = \sqrt{2}\pi \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{12} & 0 & 0 & 0 \\ 0 & \frac{1}{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{60} & 0 \\ 0 & 0 & 0 & \frac{1}{60} & 0 & -\frac{1}{30} \\ 0 & 0 & 0 & 0 & \frac{1}{30} & 0 \end{bmatrix}$$

and thus obtain matrix representations of the Koopman-von Neumann generator for both the original and whitened polynomial basis sets:

$$Q = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & -2 & 0 & 4 \\ 0 & 0 & 0 & 0 & -1 & 0 \end{bmatrix} \quad \text{and}$$

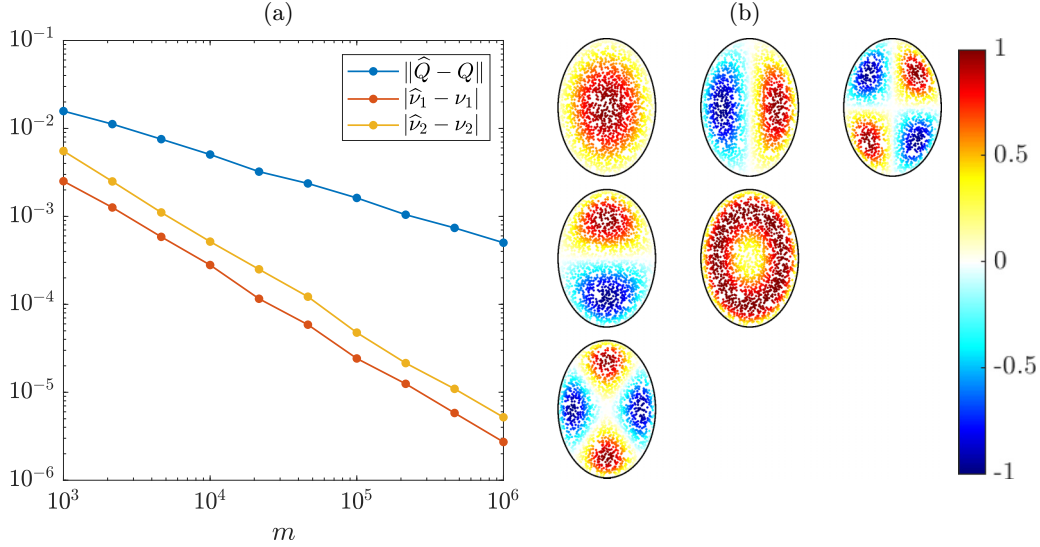


FIG. 3. (a) Convergence of the estimated matrix $\hat{Q}$ to the true matrix representation Q is approximately $\mathcal{O}(m^{-1})$, which is the expected Monte Carlo convergence rate. The observed convergence to the analytically computed eigenvalues $\nu_1 = i\sqrt{2}$ and $\nu_2 = i2\sqrt{2}$ seems to be roughly $\mathcal{O}(m^{-1})$, but there is currently no theoretical justification for the faster convergence. (b) Numerically computed eigenfunctions for $m = 2000$, cf. Fig. 1(b). We obtain only approximations of the eigenfunctions $f_\lambda(x) = f_0(x)f_{\lambda_1}(x)^{k_1}f_{\lambda_2}(x)^{k_2}$ with $k_1 + k_2 \leq 2$ due to the chosen dictionary, corresponding to the upper left triangle in Fig. 1(b).

$$\tilde{Q} = \begin{bmatrix} 0 & -\sqrt{2} & 0 & 0 & 0 & 0 \\ \sqrt{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -z_1 & -z_2 & -z_3 \\ 0 & 0 & z_1 & 0 & 0 & 0 \\ 0 & 0 & z_2 & 0 & 0 & 0 \\ 0 & 0 & z_3 & 0 & 0 & 0 \end{bmatrix}, \quad (7)$$

where $z_1, z_2, z_3 > 0$. The matrix A is already skew-symmetric since the system is Hamiltonian, but the matrix Q is not as we still have to orthonormalize the basis functions to obtain the skew-symmetric matrix $\tilde{Q}$. As the basis set defines an invariant subspace for Q , the matrix $\tilde{Q}$ can be used to exactly propagate wave functions contained in the linear span of the basis functions, and to compute select eigenvalues of Q .

Indeed, the spectrum of $\tilde{Q}$ is $\sigma(\tilde{Q}) = \{0, \pm i\sqrt{2}, \pm i2\sqrt{2}\}$, where the eigenvalue $\nu = 0$ has multiplicity two, corresponding to the first six combinations $i\sqrt{2}(k_1 - k_2)$ of the principal eigenvalues. The resulting propagator $e^{t\tilde{Q}}$ is now unitary. More precisely, given the coefficients c defining the function $\psi(x) = c^\top \phi(x)$ at time 0, we obtain the wave function $\psi(x) = (e^{t\tilde{Q}}c)^\top \phi(x)$ at time t .

We also verify the accuracy of numerical estimates of the Koopman-von Neumann generator based on the gEDMD approach outlined in Sec. III. We first uniformly sample m data points $x^{(l)}$ in Ω , define $\hat{x}^{(l)} = b(x^{(l)})$, and then compute $\hat{Q}$ and its eigenvalues and eigenvectors, which then determine the eigenfunctions.³ The numerical errors for different values

of m as well as the eigenfunctions obtained for $m = 2000$ are shown in Fig. 3.

2. Quantum circuit representation

We also use this example to show how the projected Koopman-von Neumann operator gives rise to a quantum circuit representation, which can be realized on a quantum computer. The matrix $\tilde{Q}$ in Eq. (7) and hence also $e^{t\tilde{Q}}$ consist of a block of size two and a block of size four. The 2×2 block of the propagator is

$$\begin{bmatrix} \cos(\sqrt{2}t) & -\sin(\sqrt{2}t) \\ \sin(\sqrt{2}t) & \cos(\sqrt{2}t) \end{bmatrix}$$

and can be described by the single-quantum gate $R_y(2\sqrt{2}t)$. Additionally, defining $r = \sqrt{z_1^2 + z_2^2 + z_3^2}$, $s = \sqrt{z_2^2 + z_3^2}$, and the angles

$$\theta = 2 \arctan\left(\frac{s}{z_1}\right), \quad \phi = 2 \arctan\left(\frac{z_2}{z_3}\right), \quad \beta = 2rt,$$

we can express the 4×4 block of the propagator as a quantum circuit consisting only of Pauli- X /NOT gates and controlled R_y rotations. The derivation of the circuit representation can be found in Appendix C. The overall quantum circuit is shown in Fig. 4. Note that all preprocessing steps including the computation of the Galerkin matrices G and A , the whitening transformation, and the calculation of the parameters are carried out classically. The quantum part is then restricted to the unitary time evolution. More precisely, the 2×2 block is represented by a single R_y gate acting on one qubit, while the 4×4 block requires two qubits, five controlled R_y rotations, and two Pauli- X gates. For the discretization considered here, the circuit size is constant, with the propagation time entering only through the rotation angles. However, this low resource count is a result of the specific block structure of the

³The time derivatives for the training data points could also be estimated using finite difference approximations if we only have access to trajectory data. However, since we are interested in the convergence of the method itself, we want to avoid additional numerical errors.

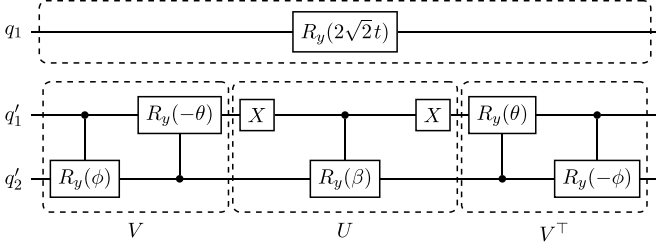


FIG. 4. Quantum circuit representation of the propagator $e^{i\tilde{Q}}$ consisting of two separate circuits. The 2×2 block is represented by one R_y gate acting on a single qubit, while the 4×4 block can be decomposed into controlled R_y gates and Pauli- X gates acting on two qubits.

considered example and does not imply a comparable scaling behavior for other discretizations. While there are algorithms to decompose arbitrary unitary matrices into quantum gates, the number of gates in general scales exponentially with the number of qubits [52]. We do not claim any quantum advantage in the present setting. The purpose of this construction is rather to illustrate that a simple classical dynamical system can be encoded within a quantum-circuit framework.

3. Random feature basis

In addition to the polynomial dictionary, we also evaluate numerical approximations of the Koopman-von Neumann generator by a more system-agnostic basis set. We introduce a basis of n tapered random Fourier features (RFFs):

$$\phi_i(x) = f_0(x) \cos(\omega_i^\top x + \theta_i). \quad (8)$$

The frequencies ω_i are drawn from a normal distribution with bandwidth σ^{-1} , which is the spectral measure of a Gaussian kernel with bandwidth σ . Moreover, $\theta_i \in [0, 2\pi]$ are uniform random phase shifts, and f_0 is the conservation law defined in Eq. (5), again ensuring that the basis functions vanish exactly on the boundary. The random features approximate the Gaussian kernel in the limit of infinite n .

In our experiments, we set the Gaussian bandwidth to $\sigma = 0.5$, and use $n = 300$ random features. In Fig. 5(a), we verify that gEDMD models trained using this basis can accurately predict the (tapered) system state variables $f_0(x)x_1$ and $f_0(x)x_2$. To this end, we learn gEDMD models using increasing amounts of uniform training data in Ω and predict the system state over one period of the system. As expected, the prediction error decreases with increasing training data size at a roughly inverse square root rate. We also track the maximal change of L^2 -norm of the predicted wave function compared to the initial condition, thereby verifying that the learned time evolution is unitary. The deviation remains on the level of floating point errors for all experiments. Finally, we confirm that the spectrum of the Koopman-von Neumann generator can be correctly identified by the RFF models. We see in Fig. 5(b) that the eigenvalues $\nu \in \{\pm i\sqrt{2}k\}$ for $0 \leq k \leq 5$ are accurately recovered. We also notice that the method is now subject to spectral pollution, but spurious eigenvalues can be reliably removed by applying the *residual score* introduced in Refs. [53,54].

B. Damped oscillator

Let us now consider the damped oscillator. We use this system as a first example of non-Hamiltonian dynamics, where the Koopman and Koopman-von Neumann generators describe genuinely different quantities. To illustrate the evolution of wave functions under the Koopman-von Neumann equation, we use a high-accuracy PDE integrator, thus minimizing numerical and statistical errors for this example. Naturally, this procedure is not scalable to higher-dimensional systems.

We decompose the domain chosen in Sec. IV with the aid of Gmsh [55] into 447 triangles and then compute the matrices A and G using FEniCS [56]. The dictionary in this case comprises piecewise linear functions. We define the initial wave function to be a superposition of two Gauss-like functions and then simulate its evolution in time. The numerical results are shown in Fig. 6. Due to the damping, the Gauss peaks,

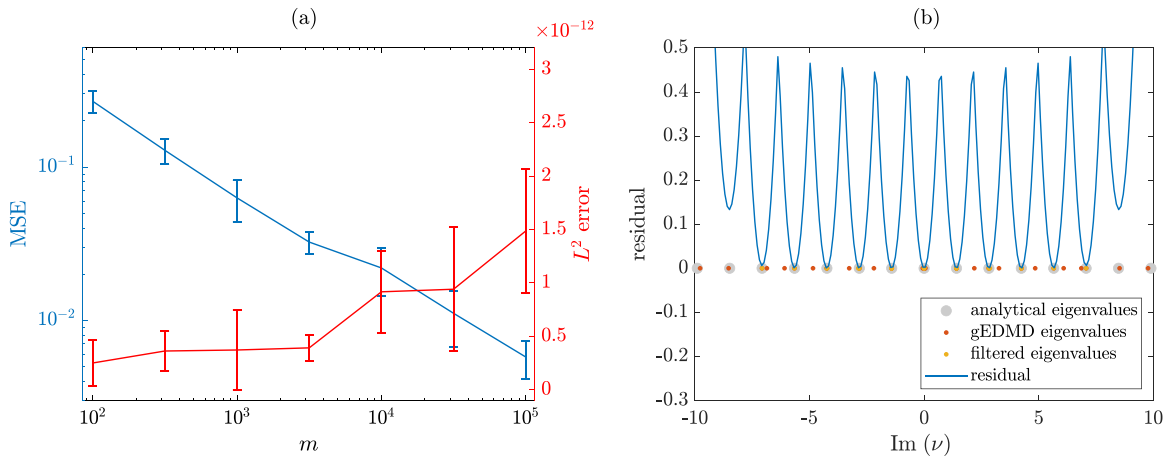


FIG. 5. (a) Mean-squared error for the state variables x_1 and x_2 predicted by gEDMD models with RFF basis (8) over one period of the undamped oscillator and L^2 error as a function of the number of training data points m . The error bars correspond to ten independent trials. (b) Comparison of the imaginary parts of the analytically computed eigenvalues (gray dots), the gEDMD eigenvalues for $m = 10^5$ (red dots), and the filtered eigenvalues (yellow dots) for which the residual score (blue curve) from Refs. [53,54] is less than 10^{-2} .

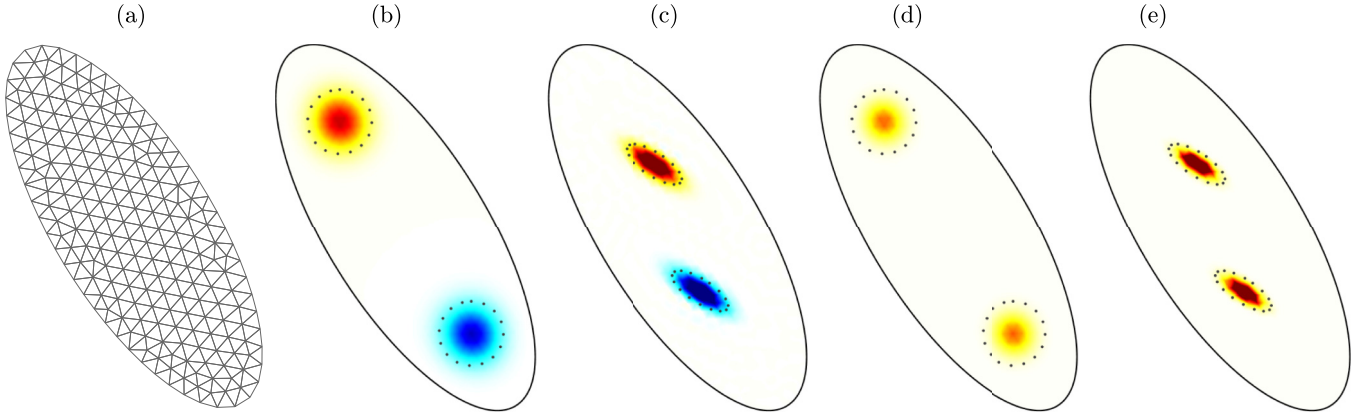


FIG. 6. (a) Gmsh triangulation of the domain $\Omega = \{x \in \mathbb{R}^2 : x_1^2 + x_1 x_2 + \frac{1}{2} x_2^2 < 1\}$ chosen for the damped oscillator. (b) Initial wave function ψ at time $t = 0$. (c) Wave function ψ at $t = \frac{1}{2}$. The black dots, shown for the sake of comparison, represent individual particles. (d) Corresponding probability density ρ at $t = 0$. (e) Density ρ at $t = \frac{1}{2}$.

as expected, quickly spiral toward the origin. At the same time, the height of the peaks increases and the bandwidth decreases. The simulation becomes unstable for larger times t as the wave function cannot be accurately resolved using the chosen basis anymore. For smaller values of the damping coefficient γ , the spiraling effect would be more pronounced. The results show that the dynamics of non-Hamiltonian systems is described accurately by wave functions evolving under the Koopman-von Neumann equation, whose propagator can, using again the linear transformation of the basis functions, be approximated by unitary matrices and hence quantum circuits.

C. Lotka-Volterra model

As a final example, we consider the nonlinear Lotka-Volterra model introduced in Sec. IV. We track the evolution of a Gaussian-like wave function centered at $\mu_0 = [\frac{1}{2}, \frac{1}{2}]^\top$ with isotropic bandwidth $\sigma_0 = \sqrt{0.02}$.

1. FEM basis

As in the previous example, we compute a high-accuracy solution of the Koopman-von Neumann equation using finite elements. We first approximate the boundary $\partial\Omega$ by B-splines and then triangulate the domain. Gmsh generates 3524 nodes and 6959 elements. The mass and stiffness matrices are again computed using FEniCS. We then simulate the Koopman-von Neumann equation from $t = 0$ to $t = 7$. The numerical results, shown in Figs. 7(a)–7(d), illustrate that the numerically computed Koopman-von Neumann wave function faithfully describes the nonlinear oscillation. From the wave function, we can recover the corresponding probability density using Born's rule.

2. Random feature basis

As for the undamped oscillator, we also compute a data-driven solution to the Koopman-von Neumann equation using a random feature basis set. We again use cosine waves with random frequencies corresponding to a Gaussian kernel with bandwidth σ and uniform phase shifts. This time, each basis

function is multiplied by an exponential tapering function

$$\eta(x) = \exp\left(-\frac{1}{kf_0(x)^2}\right),$$

to enforce the boundary conditions, where f_0 is the conservation law (6) for the Lotka-Volterra domain. The exponential tapering function decays smoothly to zero as x approaches the boundary. In the numerical experiments, we set $\sigma = 0.2$, $n = 1000$, $k = 5000$, and use $m = 50\,000$ uniformly sampled training points. Figures 7(e)–7(h) show the wave function predicted by the RFF model. We see that the nonlinear rotation around the domain is correctly captured by the RFF wave function. The RFF-based solution introduces small scale fluctuations that the FEM-based model does not display. These artifacts could likely be removed by a more customized basis set and more training data. Nevertheless, this shows that even for our quite agnostic basis set, the mean-squared error between both models at training data size $m = 100\,000$ is only on the order of 10^{-4} , see the referenced code repository for details.

VI. CONCLUSION

The Koopman-von Neumann equation describes the evolution of wave functions associated with classical dynamical systems. We analyzed the relationships between conventional transfer operators, such as the Koopman generator and Perron-Frobenius generator, and the Koopman-von Neumann framework. The main advantage is that the Koopman-von Neumann generator is skew-adjoint even for non-Hamiltonian systems and the corresponding propagator is unitary. This allows us to represent Galerkin approximations of the operator by unitary matrices, which in turn can be represented by quantum circuits. The results show that the definition of the domain and the choice of basis functions are crucial. Provided that the basis functions satisfy the homogeneous Dirichlet boundary conditions, we can compute approximations of the Koopman-von Neumann generator using either finite element methods or a generalization of gEDMD. For linear systems with conservation laws, it is possible to explicitly construct invariant subspaces and to compute eigenfunctions.

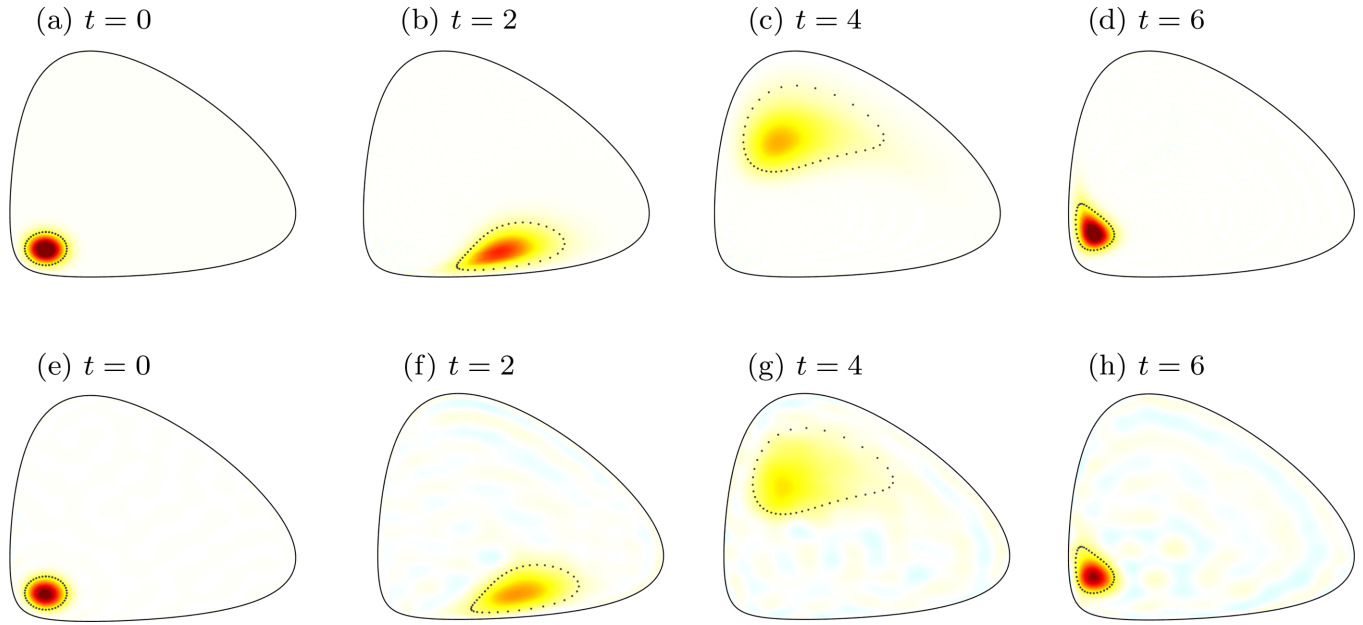


FIG. 7. Solution of the Koopman-von Neumann equation associated with the Lotka-Volterra model at different times t using an FEM basis (top row) and an RFF basis (bottom row). The initial wave function ψ is distorted by the dynamics and rotates in a counterclockwise direction. Note that the shape of the wave function will be different after the completion of one cycle since the velocities of the particles depend on the distance from the fixed point $x = [1, 1]^T$. The black dots again represent individual particles. The RFF basis leads to small numerical artifacts—even at time $t = 0$ since the initial condition needs to be approximated by tapered random Fourier features—but remains stable.

The proposed method also allows us to propagate wave functions associated with nonlinear and non-Hamiltonian systems. However, if the probability accumulates—for instance in stable fixed points—and converges to a Dirac distribution, this cannot be accurately represented by a set of finitely many basis functions, which then causes numerical instabilities.

So far, we only applied the proposed methods to low-dimensional benchmark problems. Both the finite element method and the gEDMD-based approach suffer from the curse of dimensionality. That is, for high-dimensional systems, approximating the Koopman-von Neumann operator and its eigenvalues and eigenfunctions with sufficient accuracy would require a prohibitively large dictionary. There are several commonly used techniques to mitigate this issue, which can be roughly divided into two different classes:

(1) Instead of working with an explicitly defined dictionary, kernel-based methods [30,31,57] generate potentially infinite-dimensional feature spaces and can be applied to more complex problems. Kernels can also be efficiently approximated using *random Fourier features* as shown in Ref. [58] (and the examples above). Alternatively, the dictionary can be parametrized by a deep neural network, which is then trained based on appropriate loss functions that are typically related to either the reconstruction error or implied timescales [32,33]. Randomized neural networks, where the weights of the hidden layers are randomly selected and only the output layer is optimized, can be used in the same way and are less computationally demanding [59].

(2) The second frequently used approach, which has in particular been successfully applied to high-dimensional

protein-folding problems, is to first project the data onto a dynamically relevant lower-dimensional subspace using, for example, *time-lagged independent component analysis* and to then learn the operators and their eigenvalues and eigenfunctions in the reduced collective-variable space [60–62]. This implicitly assumes that there are only a few dominant eigenvalues representing the slowest timescales, followed by a spectral gap, so that the projected dynamics retain the metastable behavior. Future work includes extending the aforementioned techniques to the Koopman-von Neumann operator and applying the resulting algorithms to higher-dimensional systems. Additionally, it will be necessary to extend the range of admissible sampling regimes beyond uniform sampling, and to derive convergence results and error bounds akin to Refs. [48,63–65].

Another open question is which steps of the overall process—defining and evaluating basis functions, constructing matrix representations of the Koopman-von Neumann generator and operator, and computing eigenvalues and eigenvectors or evolving wave functions—should be carried out on a classical computer and which on a quantum computer. Closely related to this is the question of how to define basis functions and discretization schemes that are suitable for quantum hardware. It remains to be investigated which discretization methods are most appropriate for a direct quantum implementation of the Koopman-von Neumann framework, how the resulting matrix representations can be decomposed efficiently into quantum circuits, and which classes of dynamical systems admit particularly suitable quantum representations. Our future research will also include the analysis of the scaling behavior of such approaches to identify problem

classes and system sizes for which a quantum advantage over classical methods might be possible.

ACKNOWLEDGMENTS

We thank Paul Bergold for many interesting discussions about Koopman-von Neumann mechanics and Sofie Verhees and Ashwin Nayak for their FEniCS support. S.K. was funded by a Leverhulme Trust Research Fellowship.

DATA AVAILABILITY

The code and examples that support the findings presented in this paper are publicly available [66].

APPENDIX A: COMPARISON WITH THE QECD APPROACH

The QECD approach described in Refs. [46,67,68] allows for an embedding of classical states and observables into a quantum system that faithfully represents the statistics of the classical system. Here, we show how the approach outlined in Sec. II B can be understood as an extension of QECD to systems with a nonunitary Koopman operator.

Recall from Sec. II A that observables are measurable functions $f : \Omega \mapsto \mathbb{R}$ in a suitable function space, for example in $L^\infty(\Omega)$. Observables evolve under the Koopman semigroup $f_t = \mathcal{K}^t f = f \circ \Phi^t$. Likewise, states of the system are probability densities $p \in L^1(\Omega)$, evolved in time by the Perron-Frobenius operator, i.e., $p_t = \mathcal{P}^t p$. A quantum embedding is a representation of the classical system with the following elements: (1) observables f are identified with quantum observables (self-adjoint operators) $\mathcal{A}_f$ acting on a Hilbert space H ; (2) states p are identified with density operators ρ acting on H ; (3) observables and quantum observables evolve in time in such a way that expected measurements reproduce those of the classical system.

QECD achieves these goals for an ergodic measure-preserving system on the state space Ω by (i) choosing $H = L^2(\Omega)$; (ii) representing observables as multiplication operators on H via $\mathcal{A}_f \varphi = f\varphi$; (iii) representing probability densities with density operators $\rho_p = \langle p^{\frac{1}{2}}, \cdot \rangle p^{\frac{1}{2}}$, which are pure states with wave function $\psi = p^{\frac{1}{2}}$; (iv) defining the time-evolution of the state $\rho_0 = \rho$ by the von Neumann equation

$$\rho_t = \mathcal{P}^t \rho_0 \mathcal{K}^t = e^{-it\mathcal{H}} \rho_0 e^{it\mathcal{H}},$$

where the Hamiltonian $\mathcal{H} = -i\mathcal{L}$ is the self-adjoint version of the Koopman generator. Since ρ_0 is a pure state, this reduces to solving the standard Schrödinger equation

$$i\dot{\psi}_t = \mathcal{H}\psi_t,$$

and again identifying $\rho_t = \langle \psi_t, \cdot \rangle \psi_t$. For ergodic measure-preserving systems, the Koopman generator $\mathcal{L}$ is skew-

adjoint, hence the definition $\mathcal{H} = -i\mathcal{L}$ coincides with our definition using the Koopman-von Neumann generator $\mathcal{Q}$. We can read the framework introduced in Sec. II B as using the same identification of observables and states, with the only modification that the Koopman-von Neumann generator provides the correct definition of the Hamiltonian as $\mathcal{H} = i\mathcal{Q}$ to define the quantum dynamics for a non-measure-preserving system.

In both cases, the expected value of any observable can be recovered by taking the quantum mechanical expectation of a quantum observable under the time-evolved state:

$$\begin{aligned} \text{tr}(\rho_t \mathcal{A}_f) &= \langle \psi_t, \mathcal{A}_f \psi_t \rangle = \int_{\Omega} \psi_t(x) f(x) \psi_t(x)^* dx \\ &= \int_{\Omega} |\psi_t(x)|^2 f(x) dx = \mathbb{E}^{p_t}[f] = \mathbb{E}[f(\Phi_t)]. \end{aligned}$$

The second-to-last equality, i.e., Born's rule for ψ_t , was shown in Sec. II B.

APPENDIX B: PROOF OF LEMMA II.1

- (1) We have $\mathcal{L}(fg) = b \cdot \nabla(fg) = (b \cdot \nabla f)g + f(b \cdot \nabla g) = (\mathcal{L}f)g + f(\mathcal{L}g)$.
- (2) This follows from (i) since $\mathcal{L}(f_{\lambda_1} f_{\lambda_2}) = (\lambda_1 f_{\lambda_1}) f_{\lambda_2} + f_{\lambda_1} (\lambda_2 f_{\lambda_2}) = (\lambda_1 + \lambda_2) f_{\lambda_1} f_{\lambda_2}$.
- (3) Applying the chain rule yields

$$\mathcal{L}(g \circ f_{\lambda}) = (b \cdot \nabla f_{\lambda}) g' \circ f_{\lambda} = \lambda f_{\lambda} \cdot g' \circ f_{\lambda},$$

which also implies the second statement.

- (4) Using (iii) and choosing $g(x) = x^r$, we have

$$\mathcal{L}f_{\lambda}^r = \lambda f_{\lambda} \cdot r f_{\lambda}^{r-1} = r \lambda f_{\lambda}^r.$$

- (5) It holds that

$$\begin{aligned} \mathcal{L}^*(f_0 \rho_{\mu}) &= -(b \cdot \nabla f_0) \rho_{\mu} - f_0 (b \cdot \nabla \rho_{\mu}) - \text{div}(b) f_0 \rho_{\mu} \\ &= f_0 (\mathcal{L}^* \rho_{\mu}) = \mu(f_0 \rho_{\mu}). \end{aligned}$$

- (6) The product rule implies that

$$\begin{aligned} \mathcal{L}^*(fg) &= -b \cdot \nabla(fg) - \text{div}(b)fg \\ &= -(b \cdot \nabla f)g - f(b \cdot \nabla g) - \text{div}(b)fg \\ &= (-b \cdot \nabla f - \tfrac{1}{2} \text{div}(b)f)g \\ &\quad + f(-b \cdot \nabla g - \tfrac{1}{2} \text{div}(b)g) \\ &= (\mathcal{Q}f)g + f(\mathcal{Q}g). \end{aligned}$$

- (7) Using (vi), we get $\mathcal{L}^*(\psi_{v_1} \psi_{v_2}) = (v_1 \psi_{v_1}) \psi_{v_2} + \psi_{v_1} (v_2 \psi_{v_2}) = (v_1 + v_2) \psi_{v_1} \psi_{v_2}$.
- (8) This immediately follows from the definitions of the operators since

$$\mathcal{L}^* f_{\lambda} = -b \cdot \nabla f_{\lambda} - \text{div}(b) f_{\lambda} = -\lambda f_{\lambda} - \beta f_{\lambda} = -(\lambda + \beta) f_{\lambda}.$$

The result for the Koopman-von Neumann generator follows in the same way. ■

APPENDIX C: DERIVATION OF THE QUANTUM GATE DECOMPOSITION

We consider a matrix A of the form

$$A = \begin{bmatrix} 0 & -z_1 & -z_2 & -z_3 \\ z_1 & 0 & 0 & 0 \\ z_2 & 0 & 0 & 0 \\ z_3 & 0 & 0 & 0 \end{bmatrix},$$

with $z_1, z_2, z_3 > 0$ and define $r = \sqrt{z_1^2 + z_2^2 + z_3^2}$. Since $A^3 = -r^2 A$, the exponential series for $\exp(tA)$ reduces to a combination of I , A , and A^2 , i.e.,

$$\exp(tA) = I + \frac{\sin(rt)}{r} A + \frac{1 - \cos(rt)}{r^2} A^2 = \begin{bmatrix} \cos(rt) & -\frac{\sin(rt)}{r} z_1 & -\frac{\sin(rt)}{r} z_2 & -\frac{\sin(rt)}{r} z_3 \\ \frac{\sin(rt)}{r} z_1 & 1 - \frac{1 - \cos(rt)}{r^2} z_1^2 & -\frac{1 - \cos(rt)}{r^2} z_1 z_2 & -\frac{1 - \cos(rt)}{r^2} z_1 z_3 \\ \frac{\sin(rt)}{r} z_2 & -\frac{1 - \cos(rt)}{r^2} z_1 z_2 & 1 - \frac{1 - \cos(rt)}{r^2} z_2^2 & -\frac{1 - \cos(rt)}{r^2} z_2 z_3 \\ \frac{\sin(rt)}{r} z_3 & -\frac{1 - \cos(rt)}{r^2} z_1 z_3 & -\frac{1 - \cos(rt)}{r^2} z_2 z_3 & 1 - \frac{1 - \cos(rt)}{r^2} z_3^2 \end{bmatrix}.$$

Defining $s = \sqrt{z_2^2 + z_3^2}$ and the angles

$$\theta = 2 \arctan\left(\frac{s}{z_1}\right), \quad \phi = 2 \arctan\left(\frac{z_2}{z_3}\right), \quad \beta = 2rt,$$

we can express $\exp(tA)$ as $V^\top U V$, where

$$V = CR_y^{q_2 \rightarrow q_1}(-\theta) CR_y^{q_1 \rightarrow q_2}(\phi),$$

and

$$U = (X^{q_1} \otimes I^{q_2}) CR_y^{q_1 \rightarrow q_2}(\beta) (X^{q_1} \otimes I^{q_2}).$$

Since all gates are real, V is orthogonal and hence $V^{-1} = V^+ = V^\top$. Finally, we show that the circuit indeed implements $\exp(tA)$. Define $c_\alpha = \cos(\frac{\alpha}{2})$ and $s_\alpha = \sin(\frac{\alpha}{2})$ for $\alpha \in \{\theta, \phi, \beta\}$. The controlled rotation $CR_y^{q_2 \rightarrow q_1}(-\theta)$ has the matrix representation

$$CR_y^{q_2 \rightarrow q_1}(-\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\theta & 0 & s_\theta \\ 0 & 0 & 1 & 0 \\ 0 & -s_\theta & 0 & c_\theta \end{bmatrix},$$

while

$$CR_y^{q_1 \rightarrow q_2}(\phi) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & c_\phi & -s_\phi \\ 0 & 0 & s_\phi & c_\phi \end{bmatrix}.$$

Hence, V is given by

$$V = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_\theta & s_\theta s_\phi & c_\phi s_\theta \\ 0 & 0 & c_\phi & -s_\phi \\ 0 & -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix}$$

and U by

$$U = \begin{bmatrix} c_\beta & -s_\beta & 0 & 0 \\ s_\beta & c_\beta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

A direct multiplication of the matrices gives

$$V^T U V = \begin{bmatrix} c_\beta & -c_\theta s_\beta & -s_\theta s_\phi s_\beta & -s_\theta c_\phi s_\beta \\ c_\theta s_\beta & c_\theta^2 c_\beta + s_\theta^2 & c_\theta s_\theta s_\phi (c_\beta - 1) & c_\theta s_\theta c_\phi (c_\beta - 1) \\ s_\theta s_\phi s_\beta & c_\theta s_\theta s_\phi (c_\beta - 1) & c_\phi^2 + s_\phi^2 (s_\theta^2 c_\beta + c_\theta^2) & s_\phi c_\phi (s_\theta^2 c_\beta + c_\theta^2 - 1) \\ s_\theta c_\phi s_\beta & c_\theta s_\theta c_\phi (c_\beta - 1) & s_\phi c_\phi (s_\theta^2 c_\beta + c_\theta^2 - 1) & s_\phi^2 + c_\phi^2 (s_\theta^2 c_\beta + c_\theta^2) \end{bmatrix}.$$

Using the above definitions together with elementary trigonometric identities, we have

$$c_\theta = \frac{z_1}{r}, \quad s_\theta = \frac{\sqrt{z_2^2 + z_3^2}}{r}, \quad c_\phi = \frac{z_3}{\sqrt{z_2^2 + z_3^2}}, \quad s_\phi = \frac{z_2}{\sqrt{z_2^2 + z_3^2}}.$$

Substituting these identities into the above matrix simplifies all entries to

$$V^T U V = \begin{bmatrix} c_\beta & -\frac{z_1}{r} s_\beta & -\frac{z_2}{r} s_\beta & -\frac{z_3}{r} s_\beta \\ \frac{z_1}{r} s_\beta & 1 + \frac{z_1^2}{r^2} (c_\beta - 1) & \frac{z_1 z_2}{r^2} (c_\beta - 1) & \frac{z_1 z_3}{r^2} (c_\beta - 1) \\ \frac{z_2}{r} s_\beta & \frac{z_1 z_2}{r^2} (c_\beta - 1) & 1 + \frac{z_2^2}{r^2} (c_\beta - 1) & \frac{z_2 z_3}{r^2} (c_\beta - 1) \\ \frac{z_3}{r} s_\beta & \frac{z_1 z_3}{r^2} (c_\beta - 1) & \frac{z_2 z_3}{r^2} (c_\beta - 1) & 1 + \frac{z_3^2}{r^2} (c_\beta - 1) \end{bmatrix},$$

and with $c_\beta = \cos(rt)$, $s_\beta = \sin(rt)$ this is exactly the matrix representation of $\exp(tA)$ derived above.

-
- [1] P. W. Shor, Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer, *SIAM J. Comput.* **26**, 1484 (1997).
- [2] A. Montanaro, Quantum algorithms: An overview, *npj Quantum Inf.* **2**, 15023 (2016).
- [3] L. K. Grover, A fast quantum mechanical algorithm for database search, in *Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC)* (ACM Press, New York, NY, 1996), pp. 212–219.
- [4] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, Quantum amplitude amplification and estimation, in *Quantum Computation and Information*, Contemp. Math. Vol. 305 (American Mathematical Society, Providence, RI, 2002), pp. 53–74.
- [5] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum algorithm for linear systems of equations, *Phys. Rev. Lett.* **103**, 150502 (2009).
- [6] A. M. Dalzell, S. McArdle, M. Berta, P. Bienias, C. Chen, A. Gilyén, C. T. Hann, M. J. Kastoryano, E. T. Khabiboulline, A. Kubica, G. Salton, S. Wang, and F. G. S. L. Brandão, *Quantum Algorithms: A Survey of Applications and End-to-End Complexities* (Cambridge University Press, Cambridge, UK, 2025).
- [7] D. Mauro, On Koopman-von Neumann waves, *Int. J. Mod. Phys. A* **17**, 1301 (2002).
- [8] Y. I. Bogdanov and N. A. Bogdanova, The study of classical dynamical systems using quantum theory, in *Proceedings of the International Conference on Micro- and Nano-Electronics* (SPIE, Bellingham, USA, 2014), Vol. 9440, pp. 476–484.
- [9] U. Klein, From Koopman-von Neumann theory to quantum theory, *Quantum Stud.: Math. Found.* **5**, 219 (2018).
- [10] I. Joseph, Koopman-von Neumann approach to quantum simulation of nonlinear classical dynamics, *Phys. Rev. Res.* **2**, 043102 (2020).
- [11] P. Morgan, An algebraic approach to Koopman classical mechanics, *Ann. Phys.* **414**, 168090 (2020).
- [12] Y. T. Lin, R. B. Lowrie, D. Aslangil, Y. Subaşı, and A. T. Sornborger, Challenges for quantum computation of nonlinear dynamical systems using linear representations, *arXiv:2202.02188*.
- [13] A. Lasota and M. C. Mackey, *Chaos, Fractals, and Noise: Stochastic Aspects of Dynamics*, 2nd ed., Applied Mathematical Sciences Vol. 97 (Springer, New York, NY, 1994).
- [14] M. Dellnitz and O. Junge, On the approximation of complicated dynamical behavior, *SIAM J. Numer. Anal.* **36**, 491 (1999).
- [15] I. Mezić, Spectral properties of dynamical systems, model reduction and decompositions, *Nonlinear Dyn.* **41**, 309 (2005).
- [16] M. Budišić, R. Mohr, and I. Mezić, Applied Koopmanism, *Chaos* **22**, 047510 (2012).
- [17] I. Mezić, Analysis of fluid flows via spectral properties of the Koopman operator, *Annu. Rev. Fluid Mech.* **45**, 357 (2013).
- [18] M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, A data-driven approximation of the Koopman operator: Extending dynamic mode decomposition, *J. Nonlinear Sci.* **25**, 1307 (2015).
- [19] S. Klus, P. Koltai, and C. Schütte, On the numerical approximation of the Perron-Frobenius and Koopman operator, *J. Comput. Dyn.* **3**, 51 (2016).
- [20] D. Giannakis, Data-driven spectral decomposition and forecasting of ergodic dynamical systems, *Appl. Comput. Harmon. Anal.* **47**, 338 (2019).
- [21] I. Mezić, A transfer operator approach to relativistic quantum wavefunction, *J. Phys. A: Math. Theor.* **56**, 094001 (2023).
- [22] M. Stengl, P. Gelß, S. Klus, and S. Pokutta, Existence and uniqueness of solutions of the Koopman-von Neumann equation on bounded domains, *J. Phys. A: Math. Theor.* **57**, 395302 (2024).

- [23] G. McCaul, A. Pechen, and D. I. Bondar, Entropy nonconservation and boundary conditions for Hamiltonian dynamical systems, *Phys. Rev. E* **99**, 062121 (2019).
- [24] F. Grotto, Essential self-adjointness of Liouville operator for 2D Euler point vortices, *J. Funct. Anal.* **279**, 108635 (2020).
- [25] S. Simon, R. Santagati, M. Degroote, N. Moll, M. Streif, and N. Wiebe, Improved precision scaling for simulating coupled quantum-classical dynamics, *PRX Quantum* **5**, 010343 (2024).
- [26] S. Cochran, J. Stokes, P. Jayakumar, and S. Veerapaneni, An application of continuous-variable gate synthesis to quantum simulation of classical dynamics, *AVS Quantum Sci.* **7**, 023801 (2025).
- [27] I. Novikau and I. Joseph, Quantum algorithm for the advection-diffusion equation and the Koopman-von Neumann approach to nonlinear dynamical systems, *Comput. Phys. Commun.* **309**, 109498 (2025).
- [28] P. J. Schmid, Dynamic mode decomposition of numerical and experimental data, *J. Fluid Mech.* **656**, 5 (2010).
- [29] J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz, On dynamic mode decomposition: Theory and applications, *J. Comput. Dyn.* **1**, 391 (2014).
- [30] M. O. Williams, C. W. Rowley, and I. G. Kevrekidis, A kernel-based method for data-driven Koopman spectral analysis, *J. Comput. Dyn.* **2**, 247 (2015).
- [31] S. Klus, I. Schuster, and K. Muandet, Eigendecompositions of transfer operators in reproducing kernel Hilbert spaces, *J. Nonlinear Sci.* **30**, 283 (2019).
- [32] Q. Li, F. Dietrich, E. M. Boltt, and I. G. Kevrekidis, Extended dynamic mode decomposition with dictionary learning: A data-driven adaptive spectral decomposition of the Koopman operator, *Chaos* **27**, 103111 (2017).
- [33] A. Mardt, L. Pasquali, H. Wu, and F. Noé, VAMPnets for deep learning of molecular kinetics, *Nat. Commun.* **9**, 5 (2018).
- [34] A. Mauroy and J. Goncalves, Linear identification of nonlinear systems: A lifting technique based on the Koopman operator, in *Proceedings of the IEEE 55th Conference on Decision and Control (CDC)* (IEEE, Piscataway, NJ, 2016), pp. 6500–6505.
- [35] S. Klus, F. Nüske, S. Peitz, J.-H. Niemann, C. Clementi, and C. Schütte, Data-driven approximation of the Koopman generator: Model reduction, system identification, and control, *Physica D* **406**, 132416 (2020).
- [36] L. Molgedey and H. G. Schuster, Separation of a mixture of independent signals using time delayed correlations, *Phys. Rev. Lett.* **72**, 3634 (1994).
- [37] C. Penland and P. D. Sardeshmukh, The optimal growth of tropical sea surface temperature anomalies, *J. Clim.* **8**, 1999 (1995).
- [38] F. Noé and F. Nüske, A variational approach to modeling slow processes in stochastic dynamical systems, *Multiscale Model. Simul.* **11**, 635 (2013).
- [39] S. Klus, F. Nüske, P. Koltai, H. Wu, I. Kevrekidis, C. Schütte, and F. Noé, Data-driven model reduction and transfer operator approximation, *J. Nonlinear Sci.* **28**, 985 (2018).
- [40] M. J. Colbrook, The multiverse of dynamic mode decomposition algorithms, in *Numerical Analysis Meets Machine Learning*, edited by S. Mishra and A. Townsend, Handbook of Numerical Analysis Vol. 25 (Elsevier, Amsterdam, 2024).
- [41] S. Klus, F. Nüske, and B. Hamzi, Kernel-based approximation of the Koopman generator and Schrödinger operator, *Entropy* **22**, 722 (2020).
- [42] S. Klus, F. Nüske, and S. Peitz, Koopman analysis of quantum systems, *J. Phys. A: Math. Theor.* **55**, 314002 (2022).
- [43] K. Meyer, G. Hall, and D. Offin, Introduction to Hamiltonian dynamical systems and the N-body problem, in *Applied Mathematical Sciences* Vol. 90 (Springer, New York, NY, 2008).
- [44] E. Kaiser, J. N. Kutz, and S. L. Brunton, Data-driven discovery of Koopman eigenfunctions for control, *Mach. Learn.: Sci. Technol.* **2**, 035023 (2021).
- [45] S. Jin, N. Liu, and Y. Yu, Quantum simulation of partial differential equations via Schrödingerization, *Phys. Rev. Lett.* **133**, 230602 (2024).
- [46] D. Giannakis, A. Ourmazd, P. Pfeffer, J. Schumacher, and J. Slawinska, Embedding classical dynamics in a quantum computer, *Phys. Rev. A* **105**, 052404 (2022).
- [47] A. Mauroy and I. Mezić, Global stability analysis using the eigenfunctions of the Koopman operator, *IEEE Trans. Autom. Control* **61**, 3356 (2016).
- [48] M. Korda and I. Mezić, On convergence of extended dynamic mode decomposition to the Koopman operator, *J. Nonlinear Sci.* **28**, 687 (2018).
- [49] J. J. Bramburger and G. Fantuzzi, Auxiliary functions as Koopman observables: Data-driven analysis of dynamical systems via polynomial optimization, *J. Nonlinear Sci.* **34**, 8 (2024).
- [50] J. Waldvogel, The period in the Lotka-Volterra system is monotonic, *J. Math. Anal. Appl.* **114**, 178 (1986).
- [51] Y.-A. Ma and H. Qian, A thermodynamic theory of ecology: Helmholtz theorem for Lotka-Volterra equation, extended conservation law, and stochastic predator-prey dynamics, *Proc. R. Soc. A* **471**, 20150456 (2015).
- [52] M. Plesch and Č. Brukner, Quantum-state preparation with universal gate decompositions, *Phys. Rev. A* **83**, 032302 (2011).
- [53] M. J. Colbrook, The mpEDMD algorithm for data-driven computations of measure-preserving dynamical systems, *SIAM J. Numer. Anal.* **61**, 1585 (2023).
- [54] M. J. Colbrook, L. J. Ayton, and M. Szöke, Residual dynamic mode decomposition: Robust and verified Koopmanism, *J. Fluid Mech.* **955**, A21 (2023).
- [55] C. Geuzaine and J.-F. Remacle, Gmsh: A 3-d finite element mesh generator with built-in pre-and post-processing facilities, *Int. J. Numer. Methods Eng.* **79**, 1309 (2009).
- [56] I. A. Baratta, J. P. Dean, J. S. Dokken, M. Habera, J. Hale, C. N. Richardson, M. E. Rognes, M. W. Scroggs, N. Sime, and G. N. Wells, DOLFINx: The next generation FEniCS problem solving environment, Zenodo, 2023, <https://doi.org/10.5281/zenodo.10447666>.
- [57] S. Das and D. Giannakis, Koopman spectra in reproducing kernel Hilbert spaces, *Appl. Comput. Harmon. Anal.* **49**, 573 (2020).
- [58] F. Nüske and S. Klus, Efficient approximation of molecular kinetics using random Fourier features, *J. Chem. Phys.* **159**, 074105 (2023).
- [59] M. Tabish, B. Leimkuhler, and S. Klus, How deep is your network? Deep vs. shallow learning of transfer operators, *arXiv:2509.19930*.
- [60] G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis, and F. Noé, Identification of slow molecular order parameters for Markov model construction, *J. Chem. Phys.* **139**, 015102 (2013).

- [61] F. Nüske, B. G. Keller, G. Pérez-Hernández, A. S. J. S. Mey, and F. Noé, Variational approach to molecular kinetics, *J. Chem. Theory Comput.* **10**, 1739 (2014).
- [62] C. Schütte, S. Klus, and C. Hartmann, Overcoming the timescale barrier in molecular dynamics: Transfer operators, variational principles and machine learning, *Acta Numer.* **32**, 517 (2023).
- [63] F. Nüske, S. Peitz, F. Philipp, M. Schaller, and K. Worthmann, Finite-data error bounds for Koopman-based prediction and control, *J. Nonlinear Sci.* **33**, 14 (2023).
- [64] C. Zhang and E. Zuazua, A quantitative analysis of Koopman operator methods for system identification and predictions, *C. R. Méc.* **351**, 721 (2023).
- [65] L. Llamazares-Elias, S. Llamazares-Elias, J. Latz, and S. Klus, Data-driven approximation of Koopman operators and generators: Convergence rates and error bounds, *J. Comput. Dyn.* **15**, 163 (2026).
- [66] S. Klus, F. Nüske, and P. Gelß, Code and data for “Numerical approximation of the Koopman-von Neumann equation: Operator learning and quantum computing” (2026), <https://github.com/sklus/d3s>; https://github.com/fnueske/kvn_ode.
- [67] D. C. Freeman, D. Giannakis, and J. Slawinska, Quantum mechanics for closure of dynamical systems, *Multiscale Model. Simul.* **22**, 283 (2024).
- [68] D. Giannakis and M. Montgomery, *Koopman and Transfer Operator Techniques from the Perspective of Quantum Theory* (Springer, Basel, 2026), pp. 1–35.