The Journal of Chemical Physics ARTICLE scitation.org/journal/jcp

Galerkin approximation of dynamical quantities using trajectory data

Cite as: J. Chem. Phys. 150, 244111 (2019); doi: 10.1063/1.5063730
Submitted: 30 September 2018 • Accepted: 13 May 2019 •
Published Online: 26 June 2019

Erik H. Thiede,[1,a] Dimitrios Giannakis,[2,b] Aaron R. Dinner,[1,a] and Jonathan Weare[2,b]

AFFILIATIONS

[1] Department of Chemistry and James Franck Institute, The University of Chicago, Chicago, Illinois 60637, USA
[2] Courant Institute of Mathematical Sciences, New York University, New York, New York 10012, USA

Note: This article is part of the Special Topic "Markov Models of Molecular Kinetics" in J. Chem. Phys.
[a] **Electronic addresses:** thiede@uchicago.edu and dinner@uchicago.edu
[b] **Electronic addresses:** dimitris@cims.nyu.edu and weare@nyu.edu

ABSTRACT

Understanding chemical mechanisms requires estimating dynamical statistics such as expected hitting times, reaction rates, and committors. Here, we present a general framework for calculating these dynamical quantities by approximating boundary value problems using dynamical operators with a Galerkin expansion. A specific choice of basis set in the expansion corresponds to the estimation of dynamical quantities using a Markov state model. More generally, the boundary conditions impose restrictions on the choice of basis sets. We demonstrate how an alternative basis can be constructed using ideas from diffusion maps. In our numerical experiments, this basis gives results of comparable or better accuracy to Markov state models. Additionally, we show that delay embedding can reduce the information lost when projecting the system's dynamics for model construction; this improves estimates of dynamical statistics considerably over the standard practice of increasing the lag time.

I. INTRODUCTION

Molecular dynamics simulations allow chemical mechanisms to be studied in atomistic detail. By averaging over trajectories, one can estimate dynamical statistics such as mean first-passage times or committors. These quantities are integral to chemical rate theories.[1–3] However, events of interest often occur on time scales several orders of magnitude longer than the time scales of microscopic fluctuations. In such cases, collecting chemical-kinetic statistics by integrating the system's equations of motion and directly computing averages (sample means) requires prohibitively large amounts of computational resources.

The traditional way to address this separation in time scales was through theories of activated processes.[2,4] By assuming that the kinetics are dominated by passage through a single transition state, researchers were able to obtain approximate analytical forms for reaction rates and related quantities. These expressions can be connected with microscopic simulations by evaluating contributing statistics, such as the potential of mean force and the diffusion tensor.[5–7] However, many processes involve multiple reaction pathways, such as the folding of larger proteins.[8,9] In these cases, it may not be possible in practice, or even in principle, to represent the system in a way that the assumptions underlying theories of activated processes are reasonable.

More recently, transition path sampling algorithms, which focus sampling on the pathways connecting metastable states, have been used to estimate rates.[10,11] Given such trajectories, dynamical statistics, such as committors, can be learned.[12,13] Short trajectories reaching the metastable states can be harvested efficiently, but sampling long trajectories, especially those including multiple intermediates, becomes difficult.[14,15] Another approach is to use splitting schemes, which aim to efficiently direct sampling by intelligently splitting and reweighting short trajectory segments.[16–25] Some of these methods can yield results that are exact up to statistical precision, with minimal assumptions about the dynamics.[20–25] However, the efficiency of these schemes is generally dependent on a reasonable choice of low-dimensional *collective variable* (CV) space: a projection of the system's phase space. Not only can this choice be

nonobvious,[12] but it can also be statistic specific. Moreover, starting and stopping the molecular dynamics many times based on the values of the CVs may be impractical depending on the implementation of the molecular dynamics engine and the overhead associated with computational communication.

A third approach is the construction of Markov state models (MSMs).[26–28] Here, the dynamics of the system are modeled as a discrete-state Markov chain with state-to-state transition probabilities estimated from previously sampled data. Projecting the dynamics onto a finite-dimensional model introduces a systematic bias, although this bias goes to zero in an appropriate limit of infinitely many states.[29] While MSMs were initially developed as a technique for approximating the slowest eigenmodes of a system's dynamics,[26] MSMs can also be used to calculate dynamical statistics for the study of kinetics.[30–32] Since MSM construction only requires time pairs separated by a single lag time, one has more freedom in how one generates the molecular dynamics data. In particular, if the lag time is sufficiently short, MSMs can be used to estimate rates even in the absence of full reactive trajectories. Constructing an efficient MSM requires projection onto CVs, and the systematic error in the resulting estimates can depend strongly on how they are defined. However, the CV space can generally be higher dimensional since it is only used to define Markov states.

It has been shown that calculating the system's eigenmodes with MSMs can be generalized to a basis expansion of the eigenmodes using an arbitrary basis set.[29,33,34] In this paper, we show that a similar generalization is possible for other dynamical statistics. Rather than solving eigenproblems, these quantities solve linear boundary value problems. This raises additional challenges: not only do the solutions obey specific boundary conditions, but the resulting approximations are also sensitive to the choice of lag time. We provide numerical schemes to address these difficulties.

We organize our work as follows. In Sec. II, we give background on the transition operator and review both MSMs and more general schemes for data-driven analysis of the spectrum of dynamical operators. We then continue our review with the connection between operator equations and chemical kinetics in Sec. III. In Sec. IV, we present our formalism. We discuss the choice of basis set in Sec. V and introduce a new algorithm for constructing basis sets that obey the boundary conditions our formalism requires. In Sec. VI, we show that delay embedding can recover information lost in projecting the system's dynamics onto a few degrees of freedom, negating the need for increasing the scheme's lag time to enforce Markovianity. We then demonstrate our algorithm on a collection of long trajectories of the Fip35 WW domain dataset in Sec. VII and conclude in Sec. VIII.

II. BACKGROUND

Many key quantities in chemical kinetics can be expressed through solutions to linear operator equations. Key to this formalism is the *transition operator*. We begin by assuming that the system's dynamics are given by a Markov process $\xi^{(t)}$ that is time-homogeneous, i.e., that the dynamics are time-independent. We do not put any restrictions on the nature of the system's state space. For example, if ξ is a diffusion process, the state space could be the space of real coordinates, $\mathbb{R}^n$. Similarly, for a finite-state Markov chain, it would be a finite set of configurations. We also do not assume that the dynamics are reversible or that the system is in a stationary state unless specifically noted.

The transition operator at a lag time of s is defined as

$$\mathcal{K}_s f(x) = \mathbf{E}\left[f\left(\xi^{(s)}\right)|\xi^{(0)} = x\right], \tag{1}$$

where f is a function on the state space and $\mathbf{E}$ denotes expectation. Note that due to time-homogeneity, we could just as easily have defined the transition operator with the time pair $\left(\xi^{(t)}, \xi^{(t+s)}\right)$ in place of $\left(\xi^{(0)}, \xi^{(s)}\right)$. Depending on the context in question, $\mathcal{K}_s$ may also be referred to as the Markov or Koopman operator.[35,36] We use the term transition operator as it is well established in the mathematical literature and stresses the notion that $\mathcal{K}_s$ is the generalization of the transition matrix for finite-state Markov processes. For instance, the requirement that the rows of a transition matrix sum to one generalizes to

$$\mathcal{K}_s 1 = \mathbf{E}\left[1|\xi^{(0)} = x\right] = 1. \tag{2}$$

Studying the transition operator provides, in principle, a route to analyzing the system's dynamics. Unfortunately, $\mathcal{K}_s$ is often either unknown or too complicated to be studied directly. This has motivated research into data-driven approaches that instead treat $\mathcal{K}_s$ indirectly by analyzing sampled trajectories.

A. Markov state modeling

One approach to studying chemical dynamics through the transition operator is the construction of Markov state models.[26–28] In this technique, one constructs a Markov chain on a finite state space to model the true dynamics of the system. The transition matrix of this Markov chain is then taken as a model for the true transition operator.

To construct an MSM from trajectory data, we partition the system's state space into M nonoverlapping sets. We refer to these sets as Markov states and denote them as S_i. Now, let μ be an arbitrary probability measure. If the system is initially distributed according to μ, the probability of transitioning from a set S_i to S_j after a time s is given by

$$P_{ij} = \frac{\int \mathbb{1}_{S_i}(x)\mathcal{K}_s\mathbb{1}_{S_j}(x)\mu(dx)}{\int \mathbb{1}_{S_i}(y)\mu(dy)}, \tag{3}$$

where $\mathbb{1}_{S_i}$ is the indicator function

$$\mathbb{1}_{S_i}(x) = \begin{cases} 1 & \text{for } x \text{ in } S_i \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Here, $\int f(x)\mu(dx)$ is the expectation with respect to the probability measure μ.[37] When μ has a probability density function, this integral is the same as the integral against the density, and in a finite state space, it would be a weighted average over states. This formalism lets us treat both continuous and discrete state spaces with one notation.

Because the sets S_i partition the state space, a simple calculation shows that the elements in each row of P_{ij} sum to one. P_{ij} therefore defines a transition matrix for a finite-state Markov process, where state i corresponds to the set S_i. The dynamics of this process are a model for the true dynamics, and P_{ij} is a model for the transition operator.

To build this model, we construct an estimate of P_{ij} from sampled data. A simple approach is to collect a dataset consisting of N time pairs, (X_n, Y_n). Here, the initial point X_n is drawn from μ, and Y_n is collected by starting at X_n and propagating the dynamics for time s. Note that since the choice of μ in (3) is relatively arbitrary, it can be defined implicitly through the sampling procedure. For instance, one can construct a dataset by extracting all pairs of points separated by the lag time s from a collection of trajectories; since we have assumed the dynamics are time-homogeneous, the actual physical time at which X_n was collected does not matter. We then define μ to be the measure from which our initial points $X_n^{(0)}$ were sampled. With this dataset, P_{ij} is now approximated as

$$\tilde{P}_{ij} = \frac{\sum_{n=1}^{N} \mathbb{1}_{S_j}(Y_n)\mathbb{1}_{S_i}(X_n)}{\sum_{m=1}^{N} \mathbb{1}_{S_i}(X_m)}. \tag{5}$$

Like P_{ij}, (5) defines a valid transition matrix. This is not the only approach for constructing estimates of P_{ij}. One commonly used approach modifies this procedure to ensure that $\tilde{P}_{ij}$ gives reversible dynamics. In this approach, one adds a self-consistent iteration that seeks to find the reversible transition matrix with the maximum likelihood given the data.[38,39]

The MSM approach has many attractive features. Since P_{ij} defines a valid transition matrix, the MSM defines a Markov chain that can be used as a general model for the dynamics. This model can then be simplified by merging the Markov sets to improve interpretability.[28,40,41] MSMs can also be used to estimate spectral information associated with the transition operator, such as its eigenvalues and eigenvectors, as we discuss in further detail in Sec. II B.[26,27,29,39] Finally, MSMs can be used to calculate a wide class of dynamical quantities, including committors, reaction rates, and expected hitting times.[30–32] Importantly, as constructing MSMs only requires datapoints separated by a short lag time, these long-time dynamical quantities can be evaluated using a collection of short trajectories.[42] In this paper, we focus exclusively on the latter application and consider MSMs as a technique for calculating the dynamical quantities required in rate theories.

The accuracy with which P_{ij} approximates $\mathcal{K}_s$ depends strongly on the choice of the sets S_i, and choosing good sets is a nontrivial problem in high-dimensional state spaces.[43–47] To address this issue, states are generally constructed by projecting the system's state space onto a CV space. Sets are then defined by either gridding the CV space or clustering sampled configurations based on the values of their CVs. Unfortunately, when gridding, the number of states grows exponentially with the dimension of the CV space. This is not necessarily the case for partitioning schemes based on data clustering, and the recent work in this direction appears promising.[39,48–52] In particular, recent approaches have used variational principles associated with the spectrum of $\mathcal{K}_s$ to give a quantitative notion of approximation quality across clustering procedures.[34,53–57] However, effectively clustering high-dimensional data is a nontrivial problem,[58,59] and constructing an MSM that accurately reflects the dynamics may still require knowledge of a good, relatively low-dimensional CV space.[43,44,60]

B. Data-driven solutions to eigenfunctions of dynamical operators

A related approach to characterizing chemical systems is to estimate the eigenfunctions and eigenvalues of operators associated with the system's dynamics from sampled data.[36] These separate the dynamics by time scale: eigenfunctions with larger eigenvalues correlate with the system's slower degrees of freedom. These eigenfunctions and eigenvalues can often be approximated from trajectory data, even when the transition operator is unknown. Multiple schemes that attempt this have been proposed, often independently, in different fields.[26,33,34,61–67] We refer to the family of these techniques using the umbrella term *Dynamical Operator Eigenfunction Analysis* (*DOEA*) for brevity and convenience. In this subsection, we summarize a simple DOEA scheme for the transition operator for the reader's convenience, largely following Ref. 63. We refer the reader to Ref. 36 for a discussion of other schemes.

Here, we consider the solution to the eigenproblem

$$\mathcal{K}_s\psi_l(x) = \lambda_l\psi_l(x). \tag{6}$$

We approximate ψ_l as a sum of basis functions ϕ_j with unknown coefficients a_j,

$$\psi_l(x) = \sum_{j=1}^{M} a_j\phi_j(x). \tag{7}$$

This is an example of the Galerkin approximation of (6),[26] a formalism we cover more closely in Sec. IV.

We now assume our data take the form discussed in Sec. II A. Substituting the basis expansion into (6), multiplying by $\phi_i(x)$, and taking the expectation against μ, we obtain the matrix equation

$$\sum_{j=1}^{M} K_{ij}a_j = \lambda_l \sum_{j=1}^{M} S_{ij}a_j, \tag{8}$$

where K_{ij} and S_{ij} are defined as

$$K_{ij} = \int \phi_i(x)\mathcal{K}_s\phi_j(x)\mu(dx), \tag{9}$$

$$S_{ij} = \int \phi_i(x)\phi_j(x)\mu(dx), \tag{10}$$

respectively. The matrix elements can be approximated as

$$\tilde{K}_{ij} = \frac{1}{N}\sum_{n=1}^{N} \phi_i(X_n)\phi_j(Y_n), \tag{11}$$

$$\tilde{S}_{ij} = \frac{1}{N}\sum_{n=1}^{N} \phi_i(X_n)\phi_j(X_n). \tag{12}$$

We substitute these approximations into (8) and solve for estimates of a_i and λ_l. Equation (7) can then be used to give an approximation for ψ_l.

DOEA schemes are closely linked to MSMs. Using the indicator functions from Sec. II A is mathematically equivalent to solving for the eigenfunctions of P_{ij}. Indeed, one of the first uses for MSMs

was for approximating the eigenfunctions and eigenvalues of the transition operator.[26,29]

The use of more general basis sets in DOEA allows information to be more easily extracted from high-dimensional CV spaces and gives added flexibility in algorithm design.[43,44,61,68,69] For instance, time-lagged independent component analysis (TICA) corresponds to a basis of linear functions and is commonly applied as a preprocessing step to generate CVs for MSM construction.[43,44,61] Alternatively, variational principles can be exploited to obtain the eigenfunctions of $\mathcal{K}_s$ for reversible dynamics (variational approach of conformation dynamics, VAC)[34] and, more generally, for the singular value decomposition of $\mathcal{K}_s$ (the variational approach for Markov processes, VAMP).[54,55] These principles suggest cost functions that can be used to assess how well a basis recapitulates the spectral properties of $\mathcal{K}_s$.[34,55,70] Furthermore, by directly minimizing these cost functions, one can construct nonlinear basis sets using machine learning approaches such as tensor-product algorithms or neural networks.[54,71]

While attempts have been made to define a theory of chemical dynamics purely in terms of the transition operator's eigenfunctions and eigenvalues,[53] most chemical theories require dynamical quantities, such as committors and mean first-passage times. In this work, we show that it is possible to construct estimates of these quantities using a general basis expansion. Just as DOEA schemes extend MSM estimates of spectral properties to general basis functions, our formalism generalizes the MSM estimation of the quantities used in rate theory.

III. THE GENERATOR AND CHEMICAL KINETICS

Many key quantities in chemical kinetics solve operator equations acting on functions of the state space. Below, we give a quick review of this formalism, detailing a few examples of chemically relevant quantities that can be expressed in this manner. These include statistics such as the mean first-passage time, forward and backward committors, and autocorrelation times. In particular, many of these operator equations are examples of Feynman-Kac formulas. For an in-depth treatment of this formalism, we refer the reader to Refs. 72 and 73.

In this work, we focus on analyzing data gathered from experiments or simulations. We expect the data to consist of a series of measurements collected at a fixed time interval. Therefore, rather than considering the dynamics of $\xi^{(t)}$, we will consider the dynamics of a discrete-time process $\Xi^{(t)}$ constructed by recording ξ every Δt units of time. If Δt is sufficiently small, this should not appreciably change any kinetic quantities.

In the discussion that follows, we choose to work with the generator of $\Xi^{(t)}$, defined as

$$\mathcal{L}f(x) = \frac{\mathcal{K}_{\Delta t}f(x) - f(x)}{\Delta t}, \tag{13}$$

instead of the transition operator. This makes no mathematical difference, but using $\mathcal{L}$ simplifies the presentation. We also stress that, with the exception of (24), the equations that follow hold only for a lag-time of $s = \Delta t$. For larger lag times, i.e., $s > \Delta t$, these equations only hold approximately. This is discussed further in Sec. VI.

A. Equations using the generator

We begin by considering the mean first-passage time and forward committor, two central quantities in chemical kinetics.[2,74,75] Let A and B be disjoint subsets of state space and let τ_A be the first time the system enters A,

$$\tau_A = \min\left\{t \geq 0 | \Xi^{(t)} \in A\right\}. \tag{14}$$

The *mean first-passage time* is the expectation of τ_A, conditioned on the dynamics starting at x,

$$m_A(x) = \mathbf{E}\left[\tau_A | \Xi^{(0)} = x\right]. \tag{15}$$

Note that $1/m_A(x)$ is a commonly used definition of the rate.[2] The *forward committor* is defined as the probability of entering B before A, conditioned on starting at x,

$$q_+(x) = \mathbf{P}\left[\tau_B < \tau_A | \Xi^{(0)} = x\right]. \tag{16}$$

Both of these quantities solve operator equations using the generator. The mean first-passage obeys the operator equation

$$\begin{aligned} \mathcal{L}m_A(x) &= -1 \text{ for } x \text{ in } A^c, \\ m_A(x) &= 0 \text{ for } x \text{ in } A. \end{aligned} \tag{17}$$

Here, A^c denotes the set of all state space configurations not in A. Equation (17) can be derived by conditioning on the first step of the dynamics. For all x in A^c, we have

$$\begin{aligned} m_A(x) &= \mathbf{E}\left[\tau_A | \Xi^{(0)} = x\right] \\ &= \mathbf{E}\left[m_A\left(\Xi^{(\Delta t)}\right) + \Delta t \middle| \Xi^{(0)} = x\right] \\ &= \mathbf{E}\left[m_A\left(\Xi^{(\Delta t)}\right) \middle| \Xi^{(0)} = x\right] + \Delta t \\ &= \mathcal{K}_{\Delta t} m_A(x) + \Delta t, \end{aligned}$$

where the second line follows from the time-homogeneity of Ξ. Rearranging then gives (17).

We can show that the forward committor obeys

$$\begin{aligned} \mathcal{L}q_+(x) &= 0 \text{ for } x \text{ in } (A \cup B)^c, \\ q_+(x) &= 0 \text{ for } x \text{ in } A, \\ q_+(x) &= 1 \text{ for } x \text{ in } B \end{aligned} \tag{18}$$

by similar arguments. We introduce the random variable

$$\mathbf{1}_{\tau_B < \tau_A} = \begin{cases} 1 \text{ if } \tau_B < \tau_A \\ 0 \text{ otherwise.} \end{cases} \tag{19}$$

For all x outside A and B, we can then write

$$\begin{aligned} q_+(x) &= \mathbf{E}\left[\mathbf{1}_{\tau_B < \tau_A} | \Xi^{(0)} = x\right] \\ &= \mathbf{E}\left[q_+\left(\Xi^{(\Delta t)}\right) \middle| \Xi^{(0)} = x\right] \\ &= \mathcal{K}_{\Delta t} q_+(x), \end{aligned}$$

which gives (18) on rearranging.

B. Expressions using adjoints of the generator

Additional quantities can be characterized using adjoints of the generator. We reintroduce the sampling measure μ from Sec. II and define the inner product

$$\langle u, v\rangle = \int u(x)v(x)\mu(dx). \tag{20}$$

Equipped with this inner product, the space of all functions that are square-integrable against μ forms a Hilbert space that we denote as L^2_μ. The unweighted adjoint of $\mathcal{L}$ is the operator $\mathcal{L}^\dagger$ such that for all u and v in the Hilbert space,

$$\langle \mathcal{L}^\dagger u, v\rangle = \langle u, \mathcal{L}v\rangle. \tag{21}$$

We now assume that the system has a unique stationary measure. The change of measure from μ to the stationary measure is defined as the function π such that

$$\int \mathbf{E}\left[f\left(\Xi^{(t)}\right)|\Xi^{(0)} = x\right]\pi(x)\mu(dx) = \int f(x)\pi(x)\mu(dx) \tag{22}$$

or equivalently,

$$\int \mathcal{L}f(x)\pi(x)\mu(dx) = 0 \tag{23}$$

holds for all functions f. As an example, if the dynamics are stationary at thermal equilibrium, we might have

$$\pi(x)\mu(dx) \propto e^{\frac{-H(x)}{k_B T}}\, dx,$$

where $H(x)$ is the system's Hamiltonian, T is the system's temperature, and k_B is the Boltzmann constant. However, this relation is not necessarily true for general state spaces or for nonequilibrium stationary states.

The change of measure to the stationary measure can be written as the solution to an expression with $\mathcal{L}^\dagger$. Interpreting (23) as an inner product, the definition of the adjoint implies

$$0 = \langle \pi, \mathcal{L}f\rangle = \langle \mathcal{L}^\dagger \pi, f\rangle$$

for all f, or equivalently,

$$\mathcal{L}^\dagger \pi(x) = 0. \tag{24}$$

The other equations may use weighted adjoints of $\mathcal{L}$. Let p be the change of measure from μ to another, currently unspecified measure. The *p-weighted adjoint* of $\mathcal{L}$ is the operator $\mathcal{L}^\dagger_p$ such that

$$\langle u, p\mathcal{L}v\rangle = \langle \mathcal{L}^\dagger_p u, pv\rangle. \tag{25}$$

A few manipulations show that the weighted adjoint can be expressed as

$$\mathcal{L}^\dagger_p f(x) = \frac{1}{p(x)}\mathcal{L}^\dagger(fp)(x). \tag{26}$$

This reduces to the unweighted adjoint when $p(x) = 1$.

One example of a formula that uses a weighted adjoint is a relation for the *backward committor*. The backward committor is the probability that, if the system is observed at configuration x and the system is in the stationary state, the system exited state A more recently than state B. It satisfies the equation

$$\begin{aligned} \mathcal{L}^\dagger_\pi q_-(x) &= 0 \text{ for } x \text{ in } (A\cup B)^c,\\ q_-(x) &= 1 \text{ for } x \text{ in } A,\\ q_-(x) &= 0 \text{ for } x \text{ in } B. \end{aligned} \tag{27}$$

Finally, we note that some quantities in chemical dynamics require the solution to multiple operator equations. For instance, in transition path theory[3] the *total reactive current* and *reaction rate* between A and B require evaluating the backward committor and the forward committor, followed by another application of the generator. The total reactive current from B to A is given by

$$\begin{aligned} I_{AB} = &\int q_-(x)\mathbb{1}_C(x)\mathcal{L}(\mathbb{1}_{C^c}q_+)(x)\pi(x)\mu(dx)\\ &- \int q_-(x)\mathbb{1}_{C^c}(x)\mathcal{L}(\mathbb{1}_C q_+)(x)\pi(x)\mu(dx). \end{aligned} \tag{28}$$

Here, C is a set that contains B but not A. The reaction rate constant is then given by

$$k_{AB} = \frac{I_{AB}}{\int q_-(x)\pi(x)\mu(dx)}. \tag{29}$$

We derive these expressions in Sec. S3 of the supplementary material through arguments very similar to those presented in Ref. 76.

Evaluating the *integrated autocorrelation time* (IAT) of a function requires estimating π, as well as solving an equation using the generator. For a function with $\int f(x)\pi(x)\mu(dx) = 0$, the IAT is the sum over the correlation function

$$t_f = \left(2\sum_{i=0}^{\infty}\frac{\int f(x)\mathcal{K}_{i\Delta t}f(x)\pi(x)\mu(dx)}{\int (f(x))^2\pi(x)\mu(dx)} - 1\right)\Delta t \tag{30}$$

and, using the Neumann series representation[77] of the appropriate pseudoinverse of $\mathcal{L}$, can be expressed as

$$t_f = 2\frac{\int f(x)\omega(x)\pi(x)\mu(dx)}{\int f(y)^2\pi(y)\mu(dy)} - \Delta t, \tag{31}$$

where ω is the solution to the equation

$$\mathcal{L}\omega(x) = f(x) \tag{32}$$

constrained to have $\int \omega(x)\pi(x)\mu(dx) = 0$.

Note that although the quantities mentioned above give us information about the long-time behavior of the system, the formalism introduced here only requires information over short time intervals. This suggests that solving these equations directly could lead to a numerical strategy for estimating these long-time statistics from short-time data.

IV. DYNAMICAL GALERKIN APPROXIMATION

Inspired by the theory behind DOEA and MSMs, we seek to solve the equations in Sec. III in a data-driven manner. We first note that the equations follow the general form

$$\begin{aligned} \mathcal{L}g(x) &= h(x) \text{ for } x \text{ in } D, \\ g(x) &= b(x) \text{ for } x \text{ in } D^c \end{aligned} \tag{33}$$

or

$$\begin{aligned} \mathcal{L}_p^\dagger g(x) &= h(x) \text{ for } x \text{ in } D, \\ g(x) &= b(x) \text{ for } x \text{ in } D^c. \end{aligned} \tag{34}$$

Here, D is a set in state space that constitutes the *domain*, g is the unknown solution, and h and b are known functions. If b is zero everywhere or D^c is empty, we say the problem has homogeneous boundary conditions.

If the generator and its adjoints are known, these equations can, in principle, be solved numerically.[78–80] However, this is generally not the case, and even if the operators are known, the dimension of the full state space is often too high to allow numerical solution. In our approach, we use approximations similar to (11) and (12) to estimate these quantities from trajectory data. This procedure only requires collections of short trajectories of the system and works when the dynamical operators are not known explicitly.

We explicitly derive the scheme for operator equations using the generator; the required modifications for equations using an adjoint require only slight modification, and are discussed at the end of Secs. IV B and IV C. We construct an approximation of the operator equation through the following steps:

1. *Homogenize boundary conditions:* If necessary, rewrite (33) as a problem with homogeneous boundary conditions using a guess for g.
2. *Construct a Galerkin scheme:* Approximate the solution as a sum of basis functions and convert the result of step 1 into a matrix equation.
3. *Approximate inner products with trajectory averages:* Approximate the terms in the Galerkin scheme using trajectory averages and solve for an estimate of g.

Since we use dynamical data to estimate the terms in a Galerkin approximation, we refer to our scheme as *Dynamical Galerkin Approximation* (DGA).

A. Homogenizing the boundary conditions

First, we rewrite (33) as a problem with homogeneous boundary conditions. This allows us to enforce the boundary conditions in step 2 by working within a vector space where every function vanishes at the boundary of the domain. If the boundary conditions are already homogeneous, either because b is explicitly zero or because D includes all of state space, this step can be skipped. We introduce a guess function r that is equal to b on D^c. We then rewrite (33) in terms of the difference between the guess and the true solution

$$y(x) = g(x) - r(x). \tag{35}$$

This converts (33) into a problem with homogeneous boundary conditions

$$\mathcal{L}y(x) = h(x) - \mathcal{L}r(x) \text{ for } x \text{ in } D, \tag{36}$$

$$y(x) = 0 \text{ for } x \text{ in } D^c. \tag{37}$$

A naive guess can always be constructed as

$$r^{naive}(x) = \mathbb{1}_{D^c}(x)b(x), \tag{38}$$

but if possible, one should attempt to choose r so that y can be efficiently expressed using the basis functions introduced in step 2.

B. Constructing the Galerkin scheme

We now approximate the solution of (36) and (37) via basis expansion using the formalism of the Galerkin approximation. Equation (36) implies that

$$\langle u\mathbb{1}_D, \mathcal{L}y\rangle = \langle u\mathbb{1}_D, h\rangle - \langle u\mathbb{1}_D, \mathcal{L}r\rangle \tag{39}$$

holds for all u in the Hilbert space L^2_μ. This is known as the *weak formulation* of (36).[81]

The space L^2_μ is typically infinite dimensional. Consequently, we cannot expect to ensure that (39) holds for every function in L^2_μ. We therefore attempt to solve (39) only on a finite-dimensional subspace of L^2_μ. To do this, we introduce a set of M linearly independent functions denoted as $\{\phi_1, \ldots, \phi_M\}$ that obey the homogeneous boundary conditions; we refer to these as the *basis functions*. The space of all linear combinations of the basis functions forms a subspace in L^2_μ which we call the *Galerkin subspace*, G. By construction, every function in G obeys the homogeneous boundary conditions. We now project (39) onto this subspace, giving the approximate equation

$$\langle \tilde{u}, \mathcal{L}\tilde{y}\rangle = \langle \tilde{u}, h\rangle - \langle \tilde{u}, \mathcal{L}r\rangle \tag{40}$$

for all $\tilde{u}$ in G. Here, $\tilde{y}$ is an element of G approximating y. Constructing G using a linear combination of basis functions that obey the homogeneous boundary conditions ensures that $\tilde{y}$ obeys the homogeneous boundary conditions as well. If we had constructed G using arbitrary basis functions, this would not be true. As we increase the dimensionality of G, we expect the error between y and $\tilde{y}$ to become arbitrarily small.

Since $\tilde{u}$ is in G, it can be written as a linear combination of basis functions. Consequently, if

$$\langle \phi_i, \mathcal{L}\tilde{y}\rangle = \langle \phi_i, h\rangle - \langle \phi_i, \mathcal{L}r\rangle$$

holds for all ϕ_i, then (40) holds for all $\tilde{u}$. Moreover, the construction of G implies that there exist unique coefficients a_j such that

$$\tilde{y}(x) = \sum_{j=1}^{M} a_j \phi_j(x), \tag{41}$$

enabling us to write

$$\sum_{j=1}^{M} L_{ij} a_j = h_i - r_i, \tag{42}$$

where

$$L_{ij} = \langle \phi_i, \mathcal{L}\phi_j\rangle, \tag{43}$$

$$h_i = \langle \phi_i, h\rangle, \tag{44}$$

$$r_i = \langle \phi_i, \mathcal{L}r\rangle. \tag{45}$$

If the terms in (43)–(45) are known, (42) can be solved for the coefficients a_j, and an estimate of g can be constructed as

$$\tilde{g}(x) = r(x) + \sum_{j=1}^{M} a_j \phi_j(x). \tag{46}$$

Since $\tilde{y}$ is zero on D^c and r obeys the inhomogeneous boundary conditions by construction,

$$\tilde{g} = r(x) = b(x) \text{ for } x \text{ in } D^c. \tag{47}$$

Consequently, our estimate of g obeys the boundary conditions.

A similar scheme can be constructed for equations with a weighted adjoint $\mathcal{L}_p^\dagger$ by adding one additional step to the procedure. After homogenizing the boundary conditions, we multiply both sides of (36) by p. We then proceed as before and obtain (42) with terms defined as

$$L_{ij} = \langle \phi_i, p\mathcal{L}_p^\dagger \phi_j \rangle = \langle \mathcal{L}\phi_i, p\phi_j \rangle, \tag{48}$$

$$h_i = \langle \phi_i, ph \rangle, \tag{49}$$

$$r_i = \langle \phi_i, p\mathcal{L}_p^\dagger r \rangle = \langle \mathcal{L}\phi_i, pr \rangle \tag{50}$$

instead of (43), (44), and (45), respectively.

C. Approximating inner products through Monte Carlo

Solving for a_j in (41) requires estimates of the other terms in (42). In general, these terms cannot be evaluated directly, due to the complexity of the dynamical operators and the high dimensionality of these integrals. However, we can estimate these terms using trajectory averages, in the style of the estimates in (11). Let $\rho_{\Delta t}$ be the joint probability measure of $\Xi^{(0)}$ and $\Xi^{(\Delta t)}$, such that for two sets X and Y in state space,

$$\int_{X,Y} \rho_{\Delta t}(dx, dy) = \mathbf{P}\left[\Xi^{(0)} \in X, \Xi^{(\Delta t)} \in Y\right]. \tag{51}$$

We observe that

$$\begin{aligned} \langle u, \mathcal{L}v \rangle &= \int u(x) \frac{\mathbf{E}\left[v\left(\Xi^{(\Delta t)}\right) | \Xi^{(0)} = x\right] - v(x)}{\Delta t} \mu(dx) \\ &= \int u(x) \frac{v(y) - v(x)}{\Delta t} \rho_{\Delta t}(dx, dy). \end{aligned} \tag{52}$$

We now assume that we have a dataset of the form described in Sec. II A, with a lag time of Δt. Since each pair (X_n, Y_n) is a draw from $\rho_{\Delta t}$, (52) can be approximated using the Monte Carlo estimate

$$\overline{\langle u, \mathcal{L}v \rangle} = \frac{1}{N} \sum_{n=1}^{N} u(X_n) \frac{v(Y_n) - v(X_n)}{\Delta t}. \tag{53}$$

Similarly, inner products of the form $\langle u, v \rangle$ can be estimated as

$$\overline{\langle u, v \rangle} = \frac{1}{N} \sum_{n=1}^{N} u(X_n) v(X_n). \tag{54}$$

If the Galerkin scheme arose from an equation with a weighted adjoint, evaluating the expectations in (48) and (50) may require p to be known *a priori*. However, if $p = \pi$, one can construct an estimate of π by applying the DGA framework to Eq. (24).

D. Pseudocode

The DGA procedure can thus be summarized as follows:

1. Sample N pairs of configurations (X_n, Y_n), where Y_n is the configuration resulting from propagating the system forward from X_n for time Δt.
2. Construct a set of M basis functions ϕ_i obeying the homogeneous boundary conditions and, if needed, the guess function r.
3. Estimate the terms in (42) using the expressions in Sec. IV C.
4. Solve the resulting matrix equations for the coefficients and substitute them into (46) to construct an estimate of the function of interest.

Some DGA estimates may require additional manipulation to ensure physical meaning. For instance, changes of measure and expected hitting times are nonnegative, and committors are constrained to be between zero and one. These bounds are not guaranteed to hold for estimates constructed through DGA. To correct this, we apply a simple postprocessing step and round the DGA estimate to the nearest value in the range. Alternatively, constraints on the mean of the solution [e.g., that for ω below (32)] can be applied by subtracting a constant from the estimate.

Finally, many dynamical quantities require the evaluation of additional inner products. For instance, to estimate the autocorrelation time, t_f, one must construct approximations to ω and π and set ω to have zero mean against $\pi(x)\mu(dx)$. One would then evaluate the numerator and denominator of (31) using (54).

To aid the reader in constructing estimates using this framework, we have written a Python package for creating DGA estimates.[82] This package also contains code for constructing the basis set we introduce in Sec. V. As part of the documentation, we have included Jupyter notebooks to aid the reader in reproducing the calculations in this work.

E. Connection with other schemes

As we have previously discussed, the DGA formalism is closely related to DOEA. Rather than considering the solution for a linear system, we could construct a Galerkin scheme for the eigenfunctions of $\mathcal{L}$. Since $\mathcal{L}$ and $\mathcal{K}_s$ have the same eigenfunctions, in the limit of infinite sampling and an arbitrarily good basis, this would give equivalent results to the scheme in Sec. II B. DOEA techniques have also been extended to solve (24).[83] A similar algorithm for addressing boundary conditions has also been suggested in the context of the data-driven study of partial differential equations and fluid flows.[84]

Our scheme is also closely related to Markov state modeling. Let ϕ_i be a basis set of indicator functions on disjoint sets S_i covering the state space. Under minor restrictions, applying DGA with this basis is equivalent to estimating the quantities in Sec. III with an MSM. We give a more thorough treatment in Sec. S1 of the supplementary material; here, we quickly motivate this connection by examining

(43) for this particular choice of basis. We note that we can divide both sides of (42) by $\int \phi_i(x)\mu(dx)$ without changing the solution. For this choice of basis, we would then have

$$\frac{L_{ij}}{\int \phi_i(x)\mu(dx)} = \frac{1}{\Delta t}(P - I)_{ij}, \tag{55}$$

where P is the MSM transition matrix defined in (3) and I is the identity matrix. Because of this similarity, we refer to a basis set constructed in this manner as an "MSM" basis.

V. BASIS CONSTRUCTION USING DIFFUSION MAPS

One natural route to improving the accuracy of DGA schemes is to improve the set of basis functions ϕ_i, thus reducing the error caused by projecting the operator equation onto the finite-dimensional subspace. Various approaches have been used to construct basis sets for describing dynamics in DOEA schemes.[43,44,61,68,69] However, if D^c is nonempty, these functions cannot be used in DGA. In particular, the linear basis in TICA cannot be used. Here, we provide a simple method for constructing basis functions with homogeneous boundary conditions based on the technique of diffusion maps.[85,86]

Diffusion maps are a technique shown to have success in finding global descriptions of molecular systems from high-dimensional input data.[87–92] A simple implementation proceeds by constructing the transition matrix

$$P^{\mathrm{DMAP}}_{mn} = \frac{K_\varepsilon(x_m, x_n)}{\sum_n K_\varepsilon(x_m, x_n)}, \tag{56}$$

where K_ε is a kernel function. This function decays exponentially with the distance between datapoints x_m and x_n at a rate set by ε. Multiple choices of K_ε exist; we give the algorithm used to construct the kernel in Sec. S2 of the supplementary material. The eigenvectors of P^{DMAP} with M highest positive eigenvalues were historically used to define a new coordinate system for dimensionality reduction. They can also be used as a basis set for DOEA and similar analyses.[66,68,93] Here, we extend this line of research, showing that diffusion maps can also be used to construct basis functions that obey homogeneous boundary conditions on arbitrary sets as required for use in DGA. We note that the diffusion process represented by P^{DMAP} is not intended as an approximation of the dynamics, but rather as a tool for building the basis functions ϕ_i. In particular, while the P^{DMAP} matrix is typically reversible, this imposes no reversibility constraint in the DGA scheme using the basis derived from P^{DMAP}.

To construct a basis set that obeys nontrivial boundary conditions, we first take the submatrix of P^{DMAP} such that $x_m, x_n \in D$. We then calculate the eigenvectors φ_i of this submatrix that have the M highest positive eigenvalues and take as our basis

$$\phi_i(x) = \begin{cases} \varphi_i(x) & \text{for } x \text{ in } D, \\ 0 & \text{otherwise.} \end{cases} \tag{57}$$

In addition to allowing us to define a basis set, P^{DMAP} gives a natural way of constructing guess functions that obey the boundary conditions. Since (56) is a transition matrix, it corresponds to a discrete Markov chain on the data. Therefore, we can construct guesses by solving analogs to (33) using the dynamics specified by the diffusion map. For equations using the generator, we solve the problem

$$\sum_n (P^{\mathrm{DMAP}} - I)_{mn} r(x_n) = h(x_m) \text{ for } m \text{ in } D, \tag{58}$$

$$r(x_m) = b(x_m) \text{ for } m \text{ in } D^c, \tag{59}$$

where I is the identity matrix. Here, the sum runs over all datapoints, not just those in D. The resulting estimate obeys the boundary conditions for all datapoints sampled in D^c.

Equation (58) can also be used to construct guesses for equations using weighted adjoints. In principle, one could replace P^{DMAP} with its weighted adjoint against p and solve the corresponding equation. However, r_n still obeys the boundary conditions irrespective of the weighted adjoint used. We therefore take the adjoint of P^{DMAP} with respect to its stationary measure. Since the Markov chain associated with the diffusion map is reversible,[85] P^{DMAP} is self-adjoint with respect to its stationary measure and we again solve (58). We discuss how to perform out-of-sample extension on the basis and the guess functions in Sec. S2 of the supplementary material.

To help the reader visualize a diffusion-map basis, we analyze a collection of datapoints sampled from the Müller-Brown potential,[94] scaled by 20 so that the barrier height is about 7 energy units; we set $k_B T = 1$. This potential is sampled using a Brownian particle with an isotropic diffusion coefficient of 0.1 using the BAOAB integrator for overdamped dynamics with a time step of 0.01 time units.[95] Trajectories are initialized out of the stationary measure by uniformly picking 10 000 starting locations on the interval $x \in (-2.5, 1.5), y \in (-2.5, 1.5)$. Initial points with potential energies larger than 100 are rejected and resampled to avoid numerical artifacts. Each trajectory is then constructed by simulating the dynamics for 500 steps, saving the position every 100 steps. We then define two states A and B (red and cyan dashed contours in Fig. 1, respectively) and construct the basis and guess functions required for the committor. The results, plotted in Fig. 1, demonstrate that the diffusion-map basis functions are smoothly varying with global support.

A. Basis set performance in high-dimensional CV spaces

We now test the effect of dimension on the performance of the basis set by attempting to calculate the forward committor and total reactive flux for a series of toy systems based on the model mentioned above. To be able to vary the dimensionality of the system, we add up to 18 harmonic "nuisance" degrees of freedom. Specifically,

$$U(x, y, z_3, \ldots, z_d) = U_{\mathrm{MB}}(x, y) + \sum_{l=3}^{d} z_l^2, \tag{60}$$

where U_{MB} is the scaled Müller-Brown potential discussed above. We compare our results with references computed by a grid-based scheme described in the supplementary material. Our reference for the committor is plotted in Fig. 3(a). We initialize the x and y dimensions as discussed above; the initial values of the nuisance coordinates were drawn from their marginal distributions at

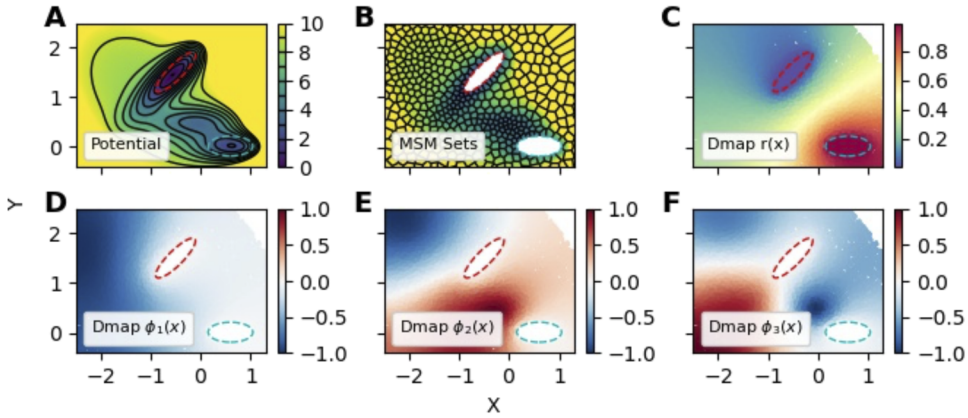

FIG. 1. Example basis and guess functions constructed by the diffusion-map basis on the scaled Müller-Brown potential. (a) The potential energy surface. Black contour lines indicate the potential energy in units of k_BT; red and cyan dotted contours indicate the boundaries of states *A* and *B*, respectively. (b) An MSM clustering with 500 sets on the domain; the color scale is the same as in (a). Each MSM basis function is one inside a cell and zero otherwise. Sets inside states A and B are not shown to emphasize the boundary conditions. (c) Scatter plot of the guess function for the committor for hitting *B* before *A*, constructed using (58). [(d)–(f)] Scatter plots of the first three basis functions constructed according to (57).

equilibrium. We then sampled the system using the same procedure as before.

Throughout this section and all subsequent numerical comparisons, we compare the diffusion-map basis with a basis of indicator functions. Since, with minor restrictions, using a basis of indicator functions is equivalent to calculating the same dynamical quantities using a MSM, we estimate committors, mean first-passage times, and stationary distributions by constructing a MSM in PyEMMA and using established formulas.[31,32,76] In general, it is not our intention to compare an optimal diffusion-map basis to an optimal MSM basis. Multiple diffusion-map and clustering schemes exist, and performing an exhaustive comparison would require comparison over multiple methods and hyperparameters. We leave such a comparison for future work and only seek to present reasonable examples of both schemes.

MSM clusters are constructed using *k*-means, as implemented in PyEMMA.[60] While MSMs are generally constructed by clustering points globally, this does not guarantee that a given clustering satisfies a specific set of boundary conditions. Consequently, we modify the set definition procedure slightly.

We first construct M clusters on the domain D and then cluster D^c separately. The number of states inside D^c is chosen so that states inside D^c have approximately the same number of samples on average as states in the domain. For the current calculation, this corresponded to approximately one state inside set *A* or *B* for every five states inside the domain; we round to a ratio of 1/5 for numerical simplicity. We note that clustering on the interior of D^c does not affect calculated committors or mean first-passage times. We use 500 basis functions for both the MSM and diffusion-map basis sets. Plots supporting this choice can be found in Sec. S5 of the supplementary material.

In modern Markov state modeling, one commonly constructs the transition matrix only over a well-connected subset of states named the active set.[39,96] We have followed this practice and excluded points outside the active set from any error analyses of the resulting MSMs. We believe this gives the MSM basis an advantage over the diffusion-map basis in our comparisons, as we are explicitly ignoring points where it fails to provide an answer and would presumably give poor results.

It is also common to ensure that the resulting matrix obeys detailed balance through a maximum likelihood procedure.[38,39]

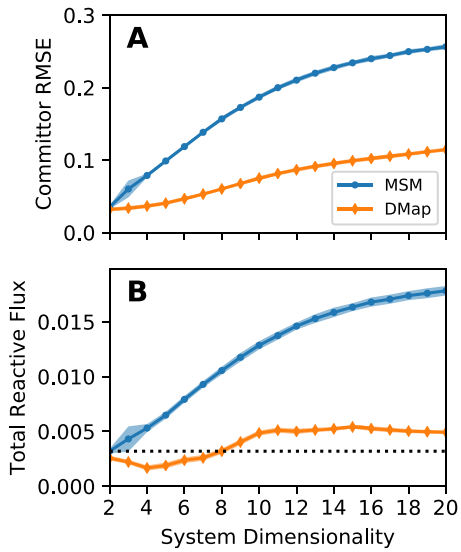

FIG. 2. Comparison of basis performance as the dimensionality of the toy system increases. (a) The average error in the forward committor between states *B* and *A* in Fig. 3 for both the MSM and the diffusion-map basis function, as a function of the number of nuisance degrees of freedom. (b) Estimated reactive flux using both the MSM and the diffusion-map basis function as a function of the same. In both plots, shading indicates the standard deviation over 30 datasets. The dotted line in (b) is the reactive flux as calculated by an accurate reference scheme.

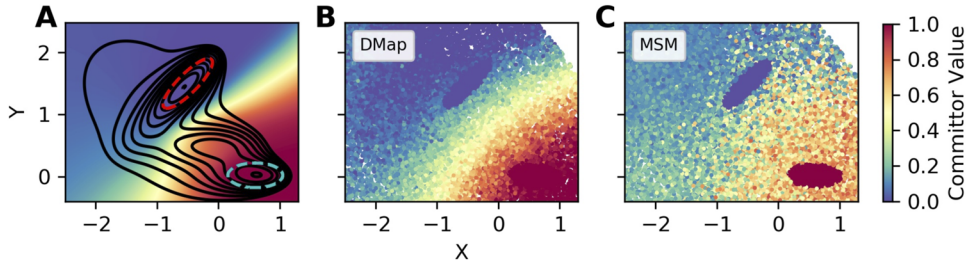

FIG. 3. Example forward committors calculated using the diffusion-map and MSM bases on a high-dimensional toy problem. The system is the same as in Fig. 1, with 18 additional nuisance dimensions. (a) Forward committor function calculated using an accurate grid-based scheme. The black lines indicate the contours of free energy in the x and y coordinates, and the red and cyan dashed contours indicate the two states. Every subsequent dimension has a harmonic potential with a force constant of 2. [(b) and (c)] Estimated forward committor constructed using the diffusion map and MSM bases, respectively.

We choose not to do this because we do not wish to assume reversibility in our formalism. Moreover, our calculations have also shown that enforcing reversibility can introduce a statistical bias that dominates the error in any estimates. We give numerical examples of this phenomenon in Sec. S6 of the supplementary material.

In Fig. 2(a), we plot root-mean-square error (RMSE) between the estimated and reference forward committors as a function of the number of nuisance degrees of freedom. While for low-dimensional systems, the MSM and the diffusion-map basis give comparable results, as we increase the dimensionality, the MSM gives increasingly worse answers. To aid in understanding these results, we plot example forward committor estimates for the 20-dimensional system in Fig. 3. We see that the diffusion-map basis manages to capture the general trends in the reference in Fig. 3(a). In contrast, the MSM basis gives considerably noisier results.

We also estimate the total reactive flux across the same dataset, setting C and C^c in (28) to be the sets on either side of the calculated isocommittor one-half surface [Fig. 2(b)]. The large errors that we observe in the reactive flux occur due to the nature of the dataset. If data were collected from a long equilibrium trajectory, it would not be necessary to estimate $\pi(x)$ separately, and we could set $\pi(x) = 1$. In that case, provided the number of MSM states was sufficient, the MSM reactive flux reverts to the direct estimation of the number of reactive trajectories per unit time. This would give an accurate reactive flux regardless of the quality of the estimated forward or backward committors.

VI. ADDRESSING PROJECTION ERROR THROUGH DELAY EMBEDDING

Our results suggest that improving basis set choice can yield DGA schemes with better accuracy in higher-dimensional CV spaces. However, even large CV spaces are considerably lower-dimensional than the system's full state space. Consequently, they may still omit key degrees of freedom needed to describe the long-time dynamics. In both MSMs and DOEA, this projection error is often addressed by increasing the lag time of the transition operator.[39,54,71,97] In the long-lag-time limit, bounds on the approximation error for DOEA show that the scheme gives the correct equilibrium averages up to projection.[29,98] However, MSMs and DOEA cannot resolve dynamics on time scales shorter than the lag time. This is reflected in existing DOEA error bounds on the relative error of the estimate of the subdominant eigenvalue, which do not vanish with increasing lag time.[98] Moreover, whereas changing the lag time does not affect the eigenfunctions in (6), the equations in Sec. III hold only for a lag time of Δt. Using a longer time is effectively making the approximation

$$\mathcal{L}f(x) \approx \frac{\mathcal{K}_s f(x) - f(x)}{s}. \tag{61}$$

This causes a systematic bias in the estimates of the dynamical quantities discussed in Sec. III. While for small lag times this bias is likely negligible, it may become large as the lag time increases. For instance, estimates of the mean first-passage time grow linearly with s as the lag time goes to infinity.[97]

Here, we propose an alternative strategy for dealing with projection error. Rather than looking at larger time lags, we use past configurations in CV space to account for contributions from the removed degrees of freedom. This idea is central to the Mori-Zwanzig formalism.[99] Here, we use *delay embedding* to include history information. Let $\zeta^{(t)}$ be the projection of $\Xi^{(t)}$ at time t. We define the delay-embedded process with d delays as

$$\theta^{(t)} = \left(\zeta^{(t)}, \zeta^{(t-\Delta t)}, \zeta^{(t-2\Delta t)}, \ldots, \zeta^{(t-d\Delta t)}\right). \tag{62}$$

Delay embedding has a long history in the study of deterministic, finite-dimensional systems, where it has been shown that delay embedding can recapture attractor manifolds up to diffeomorphism.[100,101] Weaker mathematical results have been extended to stochastic systems,[102,103] although these are not sufficient to guarantee its effectiveness in all cases.

Delay embedding has been used previously with dimensionality reduction on both experimental[104] and simulated chemical systems[105,106] and has also been used in applications of DOEA in geophysics.[66] In Refs. 66 and 107, it was argued that delay embedding can improve statistical accuracy for noise-corrupted and time-uncertain data. Other methods of augmenting the dynamical process with history information have been used in the construction of MSMs. In Ref. 108, each trajectory was augmented with a labeling variable indicating its origin state. In Ref. 97, it was suggested to write transition probabilities as a function of both the current and

the preceding MSM state. This corresponds to a specific choice of basis on a delay embedded process.

Here, we show that delay embedding can be used to improve dynamical estimates in DGA. To apply DGA to the delay-embedded process, we must extend the functions h and b in (33) and (34) to the delay-embedded space. We do this by using the value of the function on the central time point,

$$f\left(\theta^{(t)}\right) = f\left(\zeta^{(t-\lfloor d/2 \rfloor \Delta t)}\right), \tag{63}$$

where $\lfloor \ldots \rfloor$ denotes rounding down to the nearest integer. The states D and D^c in the delay-embedded space are extended similarly. One can easily show that this preserves dynamical quantities such as mean first-passage times and committors. The basis set is then constructed directly on θ, and the DGA formalism is applied as before.

We test the effect of delay embedding in the presence of projection error by constructing DGA schemes on the same system as in Sec. V and taking as our CV space only the y-coordinate. For this study, we revise our dataset to include 2000 trajectories, each sampled for 3000 time steps. While using longer trajectories changes the density such that it is closer to equilibrium, it allows us to test longer lag times and delay lengths. To ensure that our states are well-defined in this new CV space, we redefine state A to be the set $\{y > 1.15\}$, and state B to be the set $\{y < 0.15\}$. We then estimate the mean first-passage time into state A, conditioned on starting in state B at equilibrium,

$$m_{B\to A} = \frac{\int \mathbb{1}_B(x) m_A(x) \pi(x) \mu(dx)}{\int \mathbb{1}_B(x) \pi(x) \mu(dx)}.$$

We construct estimates using an MSM basis with varying lag time, an MSM basis with delay embedding, and a diffusion-map basis with delay embedding. In Fig. 4, we plot the average mean first-passage time as a function of the lag time and the trajectory length used in the delay embedding. We compare the resulting estimates with an estimate of the mean first-passage time constructed using our grid-based scheme. In addition, an implied time scale analysis for the two MSM schemes is given in Sec. S7 of the supplementary material.

The mean first-passage time estimated from the MSM basis with the lag time steadily increases as the lag time becomes longer [Fig. 4(a)], as predicted in Ref. 97. In contrast, the estimates obtained from delay embedding both converge as the delay length increases, albeit to a value slightly larger than the reference. We believe this small error is because we treat the dynamics as having a discrete time step, while the reference curve approximates the mean first-passage time for a continuous-time Brownian dynamics. In particular, the latter includes events in which the system enters and exits the target state within the duration of a discrete-time step, but such events are missing from the discrete-time dynamics.

In all three schemes, we see anomalous behavior as the length of the lag time or delay length increases. This is due to an increase in statistical error when the delay length or lag time becomes close to the length of the trajectory. If each trajectory has N datapoints, performing a delay-embedding with d delays means that each trajectory only gives $N - d$ samples. When N and d are of the same order of magnitude, this leads to increased statistical error in the estimates in Sec. IV C, to the point of making the resulting linear algebra problem ill-posed. The diffusion-map basis fluctuates to unreasonable values at long delay lengths, and the MSM basis fails completely, truncating the curve in Figs. 4(b) and 4(c). Similarly, the lagged MSM has an anomalous downturn in the average mean first-passage time near 26 time units. We give additional plots supporting this interpretation in Sec. S7 of the supplementary material.

Finally, we observe that the delay length required for the estimate to converge is substantially smaller than the mean first-passage time. This suggests that delay embedding can be effectively used on short trajectories to get estimates of long-time quantities.

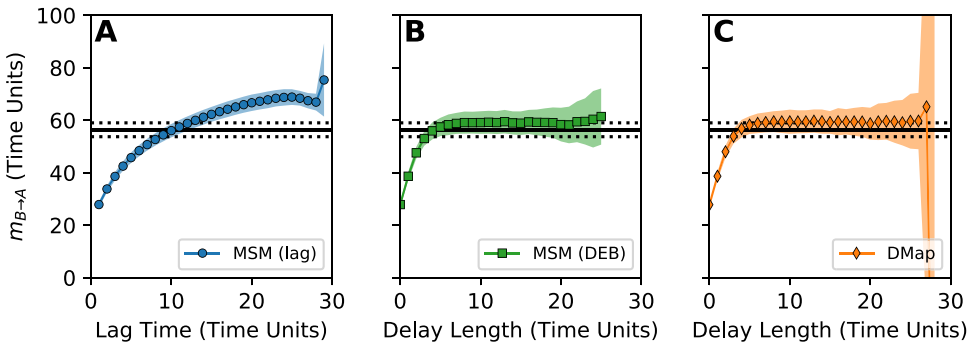

FIG. 4. Comparison of methods for dealing with the projection error in an incomplete CV space. In all subplots, we estimate the mean first-passage time from state $B = \{y < 0.15\}$ to state $A = \{y > 1.15\}$ using a DGA scheme on only the y coordinate of the Müller-Brown potential. (a) Estimate constructed using an MSM basis with increasing lag time in (61), as a function of the lag time. (b) Estimate constructed using an MSM basis, but applying delay embedding rather than increasing the lag time, as a function of the delay length. (c) Estimate constructed using the diffusion-map basis with delay embedding, as a function of the delay length. In each plot, the symbols show the mean over 30 identically constructed trajectories, and the shading indicates the standard deviation across trajectories. The black solid line is an estimate of the mean first-passage time calculated using the reference scheme in the supplementary material, and the dashed error bars represent the standard deviation of the mean first-passage time over state B.

VII. APPLICATION TO THE Fip35 WW DOMAIN

To further assess our methods, we now apply them to molecular dynamics data and seek to evaluate committors and mean first-passage times. In contrast to the simulations mentioned above, we do not have accurate reference values and cannot directly calculate the error in our estimates. Instead, we observe that both the mean first-passage time and forward committor are conditional expectations and obey the following relations:[109]

$$m_A(x) = \arg\min_{f(x)} \mathbf{E}\left[(\tau_A - f(x))^2\right],$$
$$q_+(x) = \arg\min_{f(x)} \mathbf{E}\left[(\mathbf{1}_{\tau_B<\tau_A} - f(x))^2\right].$$

This suggests a scheme for assessing the quality of our estimates. If we have access to long trajectories, each point in the trajectory has an associated sample of τ_A and $\mathbf{1}_{\tau_B<\tau_A}$. We define the two empirical cost functions

$$\mathrm{COST}_{m_A} = \frac{1}{N}\sum_{n=1}^{N}(\tilde{m}_A(x_n) - \tau_{A,n})^2, \tag{64}$$

$$\mathrm{COST}_{q_+} = \frac{1}{N}\sum_{n=1}^{N}(\tilde{q}_+(x_n) - \mathbf{1}_{\tau_B<\tau_{A,n}})^2. \tag{65}$$

Here, x_n is a collection of samples from a long trajectory, $\tau_{A,n}$ is the time from x_n to A, and $\mathbf{1}_{\tau_B<\tau_{A,n}}$ is one if the sampled trajectory next reaches B and zero if it next reaches A. The numerical estimates of the mean first-passage time and committor are written as $\tilde{m}_A$ and $\tilde{q}$, respectively. In the limit of $N \to \infty$, the true mean first-passage time and committor would minimize (64) and (65). We consequently expect lower values of our cost functions to indicate improved estimates. For a perfect estimate, however, these cost functions would not go to zero. Rather, in the limit of infinite sampling, (64) and (65) would converge to the variances of τ_A and $\mathbf{1}_{\tau_B<\tau_A}$. For the procedure to be valid, it is important that the cost estimates are not constructed using the same dataset used to build the dynamical estimates. This avoids spurious correlations between the dynamical estimate and the estimated cost.

We applied our methods to the Fip35 WW domain trajectories described by D. E. Shaw Research in Refs. 110 and 111. The dataset consists of six trajectories, each of length 100 000 ns with frames output every 0.2 ns. Each trajectory has multiple folding and unfolding events, allowing us to evaluate the empirical cost functions. To avoid correlations between the DGA estimate and the calculated cost, we perform a test/train split and divide the data into two halves. We choose three trajectories to construct our estimate and use the other three to approximate the expectations in (64) and (65). Repeating this for each possible choice of trajectories creates a total of 20 unique test/train splits.

To reduce the memory requirements in constructing the diffusion map kernel matrix, we subsampled the trajectories, keeping every 100th frame. This allowed us to test the scheme over a broad range of hyperparameters. We expect that in practical applications a finer time resolution would be used, and any additional computational expense could be offset by using landmark diffusion maps.[112]

To define the folded and unfolded states, we follow Ref. 48 and calculate $r_{\beta 1}$ and $r_{\beta 1}$, the minimum root-mean-square-displacement for each of the two β hairpins, defined as amino acids 7–23 and 18–29, respectively.[48] We define the folded configuration as having both $r_{\beta 1} < 0.2$ nm and $r_{\beta 2} < 0.13$ nm and the unfolded configuration as having 0.4 nm $< r_{\beta 1} <$ 1.0 nm and 0.3 nm $< r_{\beta 2} <$ 0.75 nm. For convenience, we refer to these states as A and B throughout this section. We then attempt to estimate the forward committor between the two states and the mean first-passage time into A using the same methods as in Sec. VI.

We take as our CVs the pairwise distances between every other α-carbon, leading to a 153-dimensional space. In previous studies, dimensionality-reduction schemes such as TICA have been applied prior to MSM construction. We choose not to do this, as we are interested in the performance of the schemes in large CV spaces. This also helps control the number of hyperparameters and algorithm design choices. Indeed, our tests suggest that, while using TICA with well-chosen hyperparameters can lead to improvements for both basis sets, the qualitative trends in our results remain unchanged. However, we think the interaction between dimensionality-reduction schemes and families of basis sets merits future investigation.

Our results are given in Fig. 5. In Figs. 5(a) and 5(b), we give the mean value of the cost for the mean first-passage time and forward committor over all test/train splits, as calculated using 200 basis functions for each algorithm. The number of basis functions was chosen to give the best result for the MSM scheme with increasing lag over any lag time, although we see only very minor differences in behavior for larger basis sets. The large standard deviations primarily reflect variation in the cost across different test/train splits, rather than any difference between the methods. This suggests the presence of large numerical noise in our results.

To get a more accurate comparison, we instead look at the expected improvement in cost between schemes for a given test/train split. To quantify whether an improvement occurs, we first determine the best parameter choice for the MSM basis with increasing lag. We estimate the cost for the MSM basis with delay embedding and for the diffusion-map basis, and calculate the difference in cost vs the lagged MSM scheme for each test/train split. We then average and calculate the standard deviation over pairs, and plot the results in Figs. 5(c)–5(f). As the difference is calculated against the best parameter choice for the lagged MSM scheme, they are intrinsically conservative: in practice, one should not expect to have the optimal lagged MSM parameters.

In our numerical experiments, we see that the diffusion map seems to give the best results for relatively short delay lengths. However, the diffusion-map basis performs progressively worse as the delay length increases. The mechanism causing this loss in accuracy requires further analysis. This tentatively suggests the use of the diffusion-map basis for datasets consisting of very short trajectories, where using long delays may be infeasible. In contrast, our results with the delay-embedded MSM basis are more ambiguous. For the mean first-passage time, we do not see significant improvement over the results from the lagged MSM results. We do see noticeable improvement in the estimated forward committor probability as the delay length increases. However, we observe that the delay lengths required to improve upon the diffusion map result are comparable in magnitude to the average time required for the trajectory to reach either the A or B states. Indeed, we only see an improvement over the diffusion-map result at a delay length of 180 ns, and we observe that

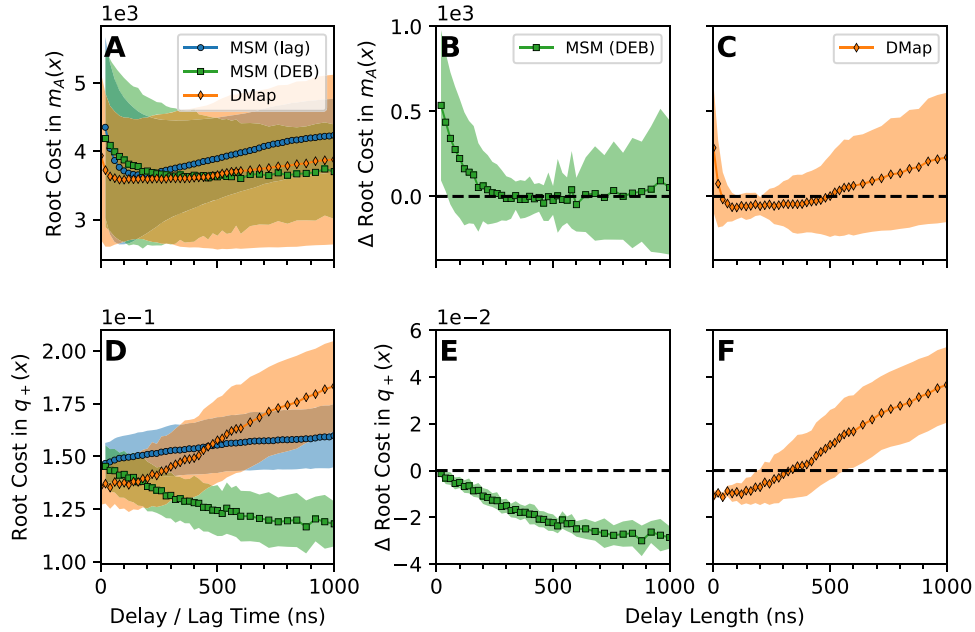

FIG. 5. Results from a DGA calculation on a dataset of six long folding and unfolding trajectories of the Fip35 WW domain. [(a) and (d)] The root cost in the mean first-passage time and forward committor, respectively, calculated using an MSM basis with increasing lag time, an MSM basis with delay embedding, and diffusion map basis with delay embedding, averaged over all test/train splits. [(b), (c), (e), and (f)] Difference in root cost relative to the best parameter choice for the estimate constructed using the MSM basis with increasing lag time. Negative values are better. (b) Difference in cost for the mean first-passage time estimated with an MSM basis with delay embedding. (c) The same as in (b) but with the diffusion map basis instead. (e) Difference in cost for the committor estimated with an MSM basis with delay embedding. (f) The same as in (e) but with the diffusion map basis instead. In all plots, the symbols are the average over test/train splits, and the shading indicates the standard deviation across test/train splits.

the longest the trajectory spends outside of both state A and state B is 223 ns. This negates any advantage of using datasets of short trajectories.

Caution is warranted in interpreting these results. We see large variances between different test/train splits, suggesting that despite having 300 μs of data in each training dataset, we are still in a relatively data-poor regime. Similarly, we cannot make an authoritative recommendation for any particular scheme for calculating dynamical quantities without further research. Such a study would not only require more simulation data but also a comparison of multiple clustering and diffusion map schemes across several hyperparameters and their interaction with various dimensionality-reduction schemes. We leave this task for future work. However, our initial results are promising, suggesting that further development of DGA schemes and basis sets is warranted.

VIII. CONCLUSIONS

In this paper, we introduce a new framework for estimating dynamical statistics from trajectory data. We express the quantity of interest as the solution to an operator equation using the generator or one of its adjoints. We then apply a Galerkin approximation, projecting the unknown function onto a finite-dimensional basis set. This allows us to approximate the problem as a system of linear equations, whose matrix elements we approximate using Monte Carlo integration on dynamical data. We refer to this framework as *Dynamical Galerkin Approximation* (DGA). These estimates can be constructed using collections of short trajectories initialized from relatively arbitrary distributions. Using a basis set of indicator functions on nonoverlapping sets recovers MSM estimates of dynamical quantities. Our work is closely related to existing work on estimating the eigenfunctions of dynamical operators in a data-driven manner.

To demonstrate the utility of alternative basis sets, we introduce a new method for constructing basis functions based on diffusion maps. Results on a toy system show that this basis has the potential to give improved results in high-dimensional CV spaces. We also combine our formalism with delay-embedding, a technique for recovering degrees of freedom omitted in constructing a CV space. Applying it to an incomplete, one-dimensional projection of our test system, we see that delay embedding can improve on the current practice of increasing the lag time of the dynamical operator.

We then applied the method to long folding trajectories of the Fip35 WW domain to study the performance of the schemes in a large CV space on a nontrivial biomolecule. Our results suggest that the diffusion-map basis gives the best performance for short delay times, giving results that are as good or better than the best time-lagged MSM parameter choice. Moreover, our results suggest that combining the MSM basis with delay embedding gives promising results, particularly, for long delay lengths. However, long delay lengths are required to see an improvement over the diffusion-map

basis, potentially negating any computational advantage in using short trajectories to estimate committors and mean first-passage times.

We believe our work raises new theoretical and algorithmic questions. Most immediately, we hope our preliminary numerical results motivate the need for new approaches to building basis sets and guess functions obeying the necessary boundary conditions. Further theoretical work is also required to assess the validity of using delay embedding in our schemes. Finally, we believe it is worth searching for connections between our work, VAC and VAMP theory,[34,54,55,70,71] and earlier approaches for learning dynamical statistics.[12,13,89] In particular, a variational reformulation of the DGA scheme would allow substantially more flexible representation of solutions. With these further developments, we believe DGA schemes have the potential to give further improved estimates of dynamical quantities for difficult molecular problems.

SUPPLEMENTARY MATERIAL

Additional theoretical and numerical support for our arguments is given in the supplementary material. We first show how MSM estimates of dynamical quantities can be derived using DGA. We then detail the specific procedure used to construct the diffusion map kernel and describe our out-of-sample extension procedure. This is followed by an adaptation of the reactive flux and transition path theory rate to discrete-time Markov chains. We then describe how we compute the reference values for dynamical quantities on the Müller-Brown potential. We next give additional plots justifying our MSM hyperparameter choices in Sec. V: we explain our choice for the number of MSM states and show that enforcing reversibility can cause substantial statistical bias in the estimates of dynamical quantities. Finally, we give additional plots examining the convergence of the delay embedded estimates in Sec. VI.

ACKNOWLEDGMENTS

This work was supported by the National Institutes of Health Award R01 GM109455 and by the Molecular Software Sciences Institute (MolSSI) Software Fellows program. Computing resources where provided by the University of Chicago Research Computing Center (RCC). The Fip35 WW domain data were provided by D.E. Shaw Research. Most of this work was completed while J.W. was a member of the Statistics Department and the James Franck Institute at the University of Chicago. We thank Charles Matthews, Justin Finkel, and Benoit Roux for helpful discussions, as well as Fabian Paul, Frank Noé, and the anonymous reviewers for their constructive feedback on earlier versions of the manuscript.

REFERENCES

[1] H. A. Kramers, Physica **7**, 284 (1940).
[2] P. Hänggi, P. Talkner, and M. Borkovec, Rev. Mod. Phys. **62**, 251 (1990).
[3] E. Vanden-Eijnden, *Computer Simulations in Condensed Matter Systems: From Materials to Chemical Biology* (Springer, 2006), Vol. 1, pp. 453–493.
[4] A. M. Berezhkovskii, A. Szabo, N. Greives, and H.-X. Zhou, J. Chem. Phys. **141**, 204106 (2014).
[5] A. Ma, A. Nag, and A. R. Dinner, J. Chem. Phys. **124**, 144911 (2006).
[6] V. Ovchinnikov, K. Nam, and M. Karplus, J. Phys. Chem. B **120**, 8457 (2016).
[7] A. Ghysels, R. M. Venable, R. W. Pastor, and G. Hummer, J. Chem. Theory Comput. **13**, 2962 (2017).
[8] A. R. Dinner and M. Karplus, J. Phys. Chem. B **103**, 7976 (1999).
[9] A. R. Dinner, A. Šali, L. J. Smith, C. M. Dobson, and M. Karplus, Trends Biochem. Sci. **25**, 331 (2000).
[10] C. Dellago, P. G. Bolhuis, F. S. Csajka, and D. Chandler, J. Chem. Phys. **108**, 1964 (1998).
[11] P. G. Bolhuis, D. Chandler, C. Dellago, and P. L. Geissler, Annu. Rev. Phys. Chem. **53**, 291 (2002).
[12] A. Ma and A. R. Dinner, J. Phys. Chem. B **109**, 6769 (2005).
[13] J. Hu, A. Ma, and A. R. Dinner, Proc. Natl. Acad. Sci. U. S. A. **105**, 4615 (2008).
[14] M. Grünwald, C. Dellago, and P. L. Geissler, J. Chem. Phys. **129**, 194101 (2008).
[15] T. R. Gingrich and P. L. Geissler, J. Chem. Phys. **142**, 234104 (2015).
[16] G. A. Huber and S. Kim, Biophys. J. **70**, 97 (1996).
[17] T. S. van Erp, D. Moroni, and P. G. Bolhuis, J. Chem. Phys. **118**, 7762 (2003).
[18] A. K. Faradjian and R. Elber, J. Chem. Phys. **120**, 10880 (2004).
[19] R. J. Allen, D. Frenkel, and P. R. ten Wolde, J. Chem. Phys. **124**, 024102 (2006).
[20] A. Warmflash, P. Bhimalapuram, and A. R. Dinner, J. Chem. Phys. **127**, 154112 (2007).
[21] E. Vanden-Eijnden and M. Venturoli, J. Chem. Phys. **131**, 044120 (2009).
[22] A. Dickson, A. Warmflash, and A. R. Dinner, J. Chem. Phys. **131**, 154104 (2009).
[23] N. Guttenberg, A. R. Dinner, and J. Weare, J. Chem. Phys. **136**, 234103 (2012).
[24] J. M. Bello-Rivas and R. Elber, J. Chem. Phys. **142**, 094102 (2015).
[25] A. R. Dinner, J. C. Mattingly, J. O. Tempkin, B. V. Koten, and J. Weare, SIAM Rev. **60**, 909 (2018).
[26] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard, J. Comput. Phys. **151**, 146 (1999).
[27] W. C. Swope, J. W. Pitera, and F. Suits, J. Phys. Chem. B **108**, 6571 (2004).
[28] V. S. Pande, K. Beauchamp, and G. R. Bowman, Methods **52**, 99 (2010).
[29] M. Sarich, F. Noé, and C. Schütte, Multiscale Model. Simul. **8**, 1154 (2010).
[30] F. Noé and S. Fischer, Curr. Opin. Struct. Biol. **18**, 154 (2008).
[31] F. Noé and J.-H. Prinz, in *An Introduction to Markov State Models and their Application to Long Timescale Molecular Simulation*, Advances in Experimental Medicine and Biology, edited by G. R. Bowman, V. S. Pande, and F. Noé (Springer, 2014), Vol. 797, Chap. 6.
[32] B. G. Keller, S. Aleksic, and L. Donati, in *Biomolecular Simulations in Drug Discovery*, edited by F. L. Gervasio and V. Spiwok (Wiley-VCH, 2019), Chap. 4.
[33] M. Weber, "Meshless methods in conformation dynamics," Ph.D. thesis, Freie Universität Berlin, 2006.
[34] F. Noé and F. Nüske, Multiscale Model. Simul. **11**, 635 (2013).
[35] T. Eisner, B. Farkas, M. Haase, and R. Nagel, *Operator Theoretic Aspects of Ergodic Theory* (Springer, 2015), Vol. 272.
[36] S. Klus, F. Nüske, P. Koltai, H. Wu, I. Kevrekidis, C. Schütte, and F. Noé, J. Nonlinear Sci. **28**, 985 (2018).
[37] P. Billingsley, *Probability and Measure* (John Wiley & Sons, 2008).
[38] G. R. Bowman, K. A. Beauchamp, G. Boxer, and V. S. Pande, J. Chem. Phys. **131**, 124101 (2009).
[39] J.-H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, M. Held, J. D. Chodera, C. Schütte, and F. Noé, J. Chem. Phys. **134**, 174105 (2011).
[40] P. Deuflhard, W. Huisinga, A. Fischer, and C. Schütte, Linear Algebra Appl. **315**, 39 (2000).
[41] S. Röblitz and M. Weber, Adv. Data Anal. Classif. **7**, 147 (2013).
[42] F. Noé, C. Schütte, E. Vanden-Eijnden, L. Reich, and T. R. Weikl, Proc. Natl. Acad. Sci. U. S. A. **106**, 19011 (2009).
[43] C. R. Schwantes and V. S. Pande, J. Chem. Theory Comput. **9**, 2000 (2013).
[44] G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis, and F. Noé, J. Chem. Phys. **139**, 015102 (2013).
[45] C. R. Schwantes, R. T. McGibbon, and V. S. Pande, J. Chem. Phys. **141**, 090901 (2014).
[46] C. Schütte and M. Sarich, Eur. Phys. J.: Spec. Top. **224**, 2445 (2015).

[47]D. Shukla, C. X. Hernández, J. K. Weber, and V. S. Pande, Acc. Chem. Res. **48**, 414 (2015).
[48]G. Berezovska, D. Prada-Gracia, and F. Rao, J. Chem. Phys. **139**, 035102 (2013).
[49]F. K. Sheong, D.-A. Silva, L. Meng, Y. Zhao, and X. Huang, J. Chem. Theory Comput. **11**, 17 (2014).
[50]Y. Li and Z. Dong, J. Chem. Inf. Model. **56**, 1205 (2016).
[51]B. E. Husic and V. S. Pande, J. Chem. Theory Comput. **13**, 963 (2017).
[52]B. E. Husic, K. A. McKiernan, H. K. Wayment-Steele, M. M. Sultan, and V. S. Pande, J. Chem. Theory Comput. **14**, 1071 (2018).
[53]J.-H. Prinz, J. D. Chodera, and F. Noé, Phys. Rev. X **4**, 011020 (2014).
[54]A. Mardt, L. Pasquali, H. Wu, and F. Noé, Nat. Commun. **9**, 5 (2018).
[55]H. Wu and F. Noé, preprint arXiv:1707.04659 (2017).
[56]W. Wang, S. Cao, L. Zhu, and X. Huang, Wiley Interdiscip. Rev.: Comput. Mol. Sci. **8**, e1343 (2018).
[57]B. E. Husic and V. S. Pande, J. Am. Chem. Soc. **140**, 2386 (2018).
[58]M. Steinbach, L. Ertöz, and V. Kumar, *New Directions in Statistical Physics* (Springer, 2004), pp. 273–309.
[59]H.-P. Kriegel, P. Kröger, and A. Zimek, ACM Trans. Knowl. Discovery Data **3**, 1 (2009).
[60]M. K. Scherer, B. Trendelkamp-Schroer, F. Paul, G. Pérez-Hernández, M. Hoffmann, N. Plattner, C. Wehmeyer, J.-H. Prinz, and F. Noé, J. Chem. Theory Comput. **11**, 5525 (2015).
[61]L. Molgedey and H. G. Schuster, Phys. Rev. Lett. **72**, 3634 (1994).
[62]H. Takano and S. Miyashita, J. Phys. Soc. Jpn. **64**, 3688 (1995).
[63]H. Hirao, S. Koseki, and H. Takano, J. Phys. Soc. Jpn. **66**, 3399 (1997).
[64]C. Schütte, F. Noé, J. Lu, M. Sarich, and E. Vanden-Eijnden, J. Chem. Phys. **134**, 204105 (2011).
[65]D. Giannakis, J. Slawinska, and Z. Zhao, *Feature Extraction: Modern Questions and Challenges* (2015), pp. 103–115.
[66]D. Giannakis, "Data-driven spectral decomposition and forecasting of ergodic dynamical systems," Appl. Comput. Harmonic Anal. (in press).
[67]M. O. Williams, I. G. Kevrekidis, and C. W. Rowley, J. Nonlinear Sci. **25**, 1307 (2015).
[68]L. Boninsegna, G. Gobbo, F. Noé, and C. Clementi, J. Chem. Theory Comput. **11**, 5947 (2015).
[69]F. Vitalini, F. Noé, and B. Keller, J. Chem. Theory Comput. **11**, 3992 (2015).
[70]F. Nüske, B. G. Keller, G. Pérez-Hernández, A. S. Mey, and F. Noé, J. Chem. Theory Comput. **10**, 1739 (2014).
[71]F. Nüske, R. Schneider, F. Vitalini, and F. Noé, J. Chem. Phys. **144**, 054105 (2016).
[72]P. Del Moral, *Feynman-Kac Formulae* (Springer, 2004).
[73]I. Karatzas and S. Shreve, *Brownian Motion and Stochastic Calculus* (Springer Science & Business Media, 2012), Vol. 113.
[74]R. Du, V. S. Pande, A. Y. Grosberg, T. Tanaka, and E. S. Shakhnovich, J. Chem. Phys. **108**, 334 (1998).
[75]P. G. Bolhuis, C. Dellago, and D. Chandler, Proc. Natl. Acad. Sci. U. S. A. **97**, 5877 (2000).
[76]P. Metzner, C. Schütte, and E. Vanden-Eijnden, Multiscale Model. Simul. **7**, 1192 (2009).
[77]K. Yosida, *Functional Analysis* (Springer-Verlag, New York, Berlin, 1971).
[78]M. Lapelosa and C. F. Abrams, Comput. Phys. Commun. **184**, 2310 (2013).
[79]R. Lai and J. Lu, Multiscale Model. Simul. **16**, 710 (2018).
[80]Y. Khoo, J. Lu, and L. Ying, Res. Math. Sci. **6**, 1 (2018).
[81]L. Evans, *Partial Differential Equations* (Orient Longman, 1998).
[82]E. Thiede, PyEDGAR, https://github.com/ehthiede/PyEDGAR/, 2018.
[83]H. Wu, F. Nüske, F. Paul, S. Klus, P. Koltai, and F. Noé, J. Chem. Phys. **146**, 154104 (2017).
[84]K. K. Chen, J. H. Tu, and C. W. Rowley, J. Nonlinear Sci. **22**, 887 (2012).
[85]R. R. Coifman and S. Lafon, Appl. Comput. Harmonic Anal. **21**, 5 (2006).
[86]T. Berry and J. Harlim, Appl. Comput. Harmonic Anal. **40**, 68 (2016).
[87]A. L. Ferguson, A. Z. Panagiotopoulos, P. G. Debenedetti, and I. G. Kevrekidis, Proc. Natl. Acad. Sci. U. S. A. **107**, 13597 (2010).
[88]M. A. Rohrdanz, W. Zheng, M. Maggioni, and C. Clementi, J. Chem. Phys. **134**, 124116 (2011).
[89]W. Zheng, B. Qi, M. A. Rohrdanz, A. Caflisch, A. R. Dinner, and C. Clementi, J. Phys. Chem. B **115**, 13065 (2011).
[90]A. L. Ferguson, A. Z. Panagiotopoulos, I. G. Kevrekidis, and P. G. Debenedetti, Chem. Phys. Lett. **509**, 1 (2011).
[91]A. W. Long and A. L. Ferguson, J. Phys. Chem. B **118**, 4228 (2014).
[92]S. B. Kim, C. J. Dsilva, I. G. Kevrekidis, and P. G. Debenedetti, J. Chem. Phys. **142**, 085101 (2015).
[93]T. Berry, D. Giannakis, and J. Harlim, Phys. Rev. E **91**, 032915 (2015).
[94]K. Müller and L. D. Brown, Theor. Chim. Acta **53**, 75 (1979).
[95]B. Leimkuhler and C. Matthews, Appl. Math. Res. Express **2013**, 34 (2012).
[96]K. A. Beauchamp, G. R. Bowman, T. J. Lane, L. Maibaum, I. S. Haque, and V. S. Pande, J. Chem. Theory Comput. **7**, 3412 (2011).
[97]E. Suárez, J. L. Adelman, and D. M. Zuckerman, J. Chem. Theory Comput. **12**, 3473 (2016).
[98]N. Djurdjevac, M. Sarich, and C. Schütte, Multiscale Model. Simul. **10**, 61 (2012).
[99]R. Zwanzig, *Nonequilibrium Statistical Mechanics* (Oxford University Press, 2001).
[100]F. Takens, Lect. Notes Math. **898**, 366 (1981).
[101]D. Aeyels, SIAM J. Control Optim. **19**, 595 (1981).
[102]M. R. Muldoon, D. S. Broomhead, J. P. Huke, and R. Hegger, Dyn. Stab. Syst. **13**, 175 (1998).
[103]J. Stark, D. Broomhead, M. Davies, and J. Huke, Nonlinear Anal.: Theory, Methods Appl. **30**, 5303 (1997).
[104]T. Berry, J. R. Cressman, Z. Greguric-Ferencek, and T. Sauer, SIAM J. Appl. Dyn. Syst. **12**, 618 (2013).
[105]J. Wang and A. L. Ferguson, Phys. Rev. E **93**, 032412 (2016).
[106]J. Wang and A. L. Ferguson, J. Phys. Chem. B **122**, 11931 (2018).
[107]R. Fung, A. M. Hanna, O. Vendrell, S. Ramakrishna, T. Seideman, R. Santra, and A. Ourmazd, Nature **532**, 471 (2016).
[108]E. Suarez, S. Lettieri, M. C. Zwier, C. A. Stringer, S. R. Subramanian, L. T. Chong, and D. M. Zuckerman, J. Chem. Theory Comput. **10**, 2658 (2014).
[109]R. Durrett, *Probability: Theory and Examples* (Cambridge University Press, 2010).
[110]D. E. Shaw, P. Maragakis, K. Lindorff-Larsen, S. Piana, R. O. Dror, M. P. Eastwood, J. A. Bank, J. M. Jumper, J. K. Salmon, Y. Shan, and W. Wriggers, Science **330**, 341 (2010).
[111]S. Piana, K. Sarkar, K. Lindorff-Larsen, M. Guo, M. Gruebele, and D. E. Shaw, J. Mol. Biol. **405**, 43 (2011).
[112]A. W. Long and A. L. Ferguson, Appl. Comput. Harmonic Anal. **47**, 190 (2019).