Appl. Comput. Harmon. Anal. 64 (2023) 62–101

Contents lists available at ScienceDirect

Applied and Computational Harmonic Analysis

journal homepage: www.elsevier.com/locate/acha

Computing committors in collective variables via Mahalanobis diffusion maps

Luke Evans [a,*], Maria K. Cameron [a], Pratyush Tiwary [b]

[a] Department of Mathematics, University of Maryland, College Park, MD 20742, USA
[b] Department of Chemistry and Biochemistry, University of Maryland, College Park, MD 20742, USA

ARTICLE INFO

Article history:
Received 22 August 2021
Received in revised form 23 September 2022
Accepted 2 January 2023
Available online 6 January 2023
Communicated by Amit Singer

Dataset link: https://github.com/aevans1/targetmeasure-mmap

Keywords:
Diffusion map
Collective variables
Committor
Anisotropic diffusion
Alanine dipeptide
Lennard-Jones

ABSTRACT

The study of rare events in molecular and atomic systems such as conformal changes and cluster rearrangements has been one of the most important research themes in chemical physics. Key challenges are associated with long waiting times rendering molecular simulations inefficient, high dimensionality impeding the use of PDE-based approaches, and the complexity or breadth of transition processes limiting the predictive power of asymptotic methods. Diffusion maps are promising algorithms to avoid or mitigate all these issues. We adapt the diffusion map with Mahalanobis kernel proposed by Singer and Coifman (2008) for the SDE describing molecular dynamics in collective variables in which the diffusion matrix is position-dependent and, unlike the case considered by Singer and Coifman, is not associated with a diffeomorphism. We offer an elementary proof showing that one can approximate the generator for this SDE discretized to a point cloud via the Mahalanobis diffusion map. We use it to calculate the committor functions in collective variables for two benchmark systems: alanine dipeptide, and Lennard-Jones-7 in 2D. For validating our committor results, we compare our committor functions to the finite-difference solution or by conducting a "committor analysis" as used by molecular dynamics practitioners. We contrast the outputs of the Mahalanobis diffusion map with those of the standard diffusion map with isotropic kernel and show that the former gives significantly more accurate estimates for the committors than the latter.

1. Introduction

1.1. Motivation

Molecular simulation commonly deals with high-dimensional systems that reside in stable states over very large timescales and transition quickly between these states on extremely small scales. These transitions, rare events such as protein folding or conformational changes in a molecule, are crucial to molecular

* Corresponding author.
E-mail addresses: evansal@umd.edu (L. Evans), mariakc@umd.edu (M.K. Cameron), ptiwary@umd.edu (P. Tiwary).

https://doi.org/10.1016/j.acha.2023.01.001
1063-5203/

simulations but difficult to characterize due to the timescale gap. Transition path theory (TPT) is a mathematical framework for direct study of rare transitions in stochastic systems, and it is particularly utilized for metastable systems arising in molecular dynamics (MD) [1]. The key function of TPT is the committor function, a mathematically well-defined reaction coordinate with which one can compute reaction channels and expected transition times between a reactant state A and a product state B in the state space. However, the committor is the solution to an elliptic PDE that can be solved using finite difference or finite element methods only in low dimensions. As a result, the typical use of transition path theory for high-dimensional systems consists of finding an estimate for a zero-temperature asymptotic transition path and then using techniques like umbrella sampling to access the committor [2,3]. While this approach is viable, it relies on the assumption that the transition process is localized to a narrow tube around the found path. In practice, the transition process may be broad and complex, and the found asymptotic path may give a poor prediction.

One can utilize intrinsic dimensionality of the system that is typically much lower than $3N_a$, where N_a is the number of atoms, and analyze transition paths in terms of *collective variables* [4,5]. However, a large number of internal coordinates such as contact distances and dihedral angles may be required for a proper representation of a biomolecule [6–9]. Hence, one still may be unable to leverage traditional mesh-based PDE solvers to even dimensionally-reduced data given either in physics-informed or machine-learned collective variables.

1.2. An overview

Meshless approaches to solving the committor PDE discretize it to point clouds obtained from MD simulations. The two most promising approaches utilize neural networks [10–12] and/or diffusion maps [13]. The approach based on neural networks is more straightforward in its implementation, while the one based on diffusion maps is more interpretable, visual, and intuitive, and we will focus on it in this work. We also would like to acknowledge the work of Lai and Lu (2018) [14] on computing committors discretized to point clouds. They proposed and advocated the "local mesh" algorithm and highlighted its advantage over an approach utilizing diffusion maps. Contrary to diffusion maps, the "local mesh" does not require the input data to be sampled from an invariant distribution, which is indeed a very appealing feature. However, the "local mesh" implementation only considers SDEs with constant noise and has no theoretical guarantees of convergence, while we found diffusion maps simple, robust, reliable, and deserving further development.

The diffusion map algorithm introduced in 2006 in the seminal work by Coifman and Lafon [15] is a widely used manifold learning algorithm. Like its predecessors such as locally linear embedding [16], isomap [17], and the Laplacian eigenmap [18], diffusion map relies on the assumption that the dataset lies in the vicinity of a certain low-dimensional manifold, while the dimension of the ambient space can be high. Importantly, this manifold does not need to be known a priori. Diffusion map inherits the use of a kernel for learning the local geometry from its predecessors and upgrades it with the remarkable ability to approximate a class of differential operators discretized to the dataset. These include the backward Kolmogorov operator (a.k.a. the generator) needed for computing the committor function.

More specifically, the standard diffusion map with isotropic Gaussian kernel [15] yields an approximation to the backward Kolmogorov operator if the input data comes from an Ito diffusion process with a gradient drift and an isotropic additive noise. The problem of finding the committor then reduces to solving a system of linear algebraic equations of a manageable size. This strategy would be suitable for MD data in the original $\mathbb{R}^{3N_a}$-dimensional space of atomic positions, and has been utilized heavily in previous work [7,13, 19–22]. As mentioned above, biophysicists prefer to keep track of collective variables rather than positions of atoms, as it lowers the dimension and gives more useful information about the molecular configuration. Unfortunately, the transformation to collective variables induces anisotropy and position-dependence on the noise term [1,4,5,23]. The resulting SDE describing the dynamics in collective variables has a non-gradient drift and a multiplicative anisotropic noise:

$$dx_t = [-M(x_t)\nabla F(x_t) + \nabla \cdot M(x_t)]\, dt + \sqrt{2\beta^{-1}} M^{1/2}(x_t) dw_t. \tag{1}$$

Here, the *diffusion matrix* $M(x)$, is a symmetric positive definite matrix function, $\nabla F(x)$ is the *mean force*, the gradient of the *free energy*, and dw is the increment of the standard multidimensional Brownian motion. The definitions and the computation of $M(x)$ and $\nabla F(x)$ are detailed in Appendix A.

1.3. The goal and a brief summary of main results

The goal of the present work is to extend the diffusion map algorithm for computing the committor functions for data sampled from the invariant density of SDE (1) and prove theoretical guarantees for its correctness. We emphasize that our application of diffusion maps is not for learning order parameters or doing dimensional reduction as in many previous works, e.g. [7,19,20,22]. We assume that we already have a dimensionally reduced system due to the use of collective variables. Instead, we are going to approximate the generator of SDE (1) by means of diffusion maps and use it to compute the committor.

If the diffusion matrix $M(x)$ in SDE (1) arose from a diffeomorphism, we could straightforwardly apply the diffusion map with the Mahalanobis kernel introduced by Singer and Coifman in 2008 [24]. However, the transformation to collective variables is not a diffeomorphism. Typically, the number of collective variables is much smaller than $3N_a$ and the computation of collective variables involves an averaging with respect to the invariant probability density. As a result, all that can be guaranteed is that the diffusion matrix $M(x)$ in (1) is symmetric positive definite. Furthermore, it is known that not every symmetric positive definite matrix function $M(x)$ is decomposable to $M(x) = J(x)J^\top(x)$ where $J(x)$ is the Jacobian matrix for some diffeomorphism [25], as required for the formalism introduced in [24].

Our results are the following:

- We offer an elementary proof showing that a smooth symmetric positive definite $d \times d$ matrix function defined in an open set Ω is not necessarily decomposable as $JJ^\top(x)$ where $J(x)$ is the Jacobian matrix of some smooth vector-function $f(x)$ in Ω. Moreover, we establish a criterion for the existence of such a decomposition for a special class of 2×2 matrix functions of the form $M(x) = m^2(x)I_{2\times 2}$ where $m(x)$ is a nonnegative twice continuously differentiable function in a simply connected open set $\Omega \subset \mathbb{R}^2$.
- We prove a theorem establishing a family of differential operators approximated by the family of diffusion maps with Mahalanobis kernel with an arbitrary smooth symmetric positive definite matrix function $M(x)$ parametrized by the renormalization parameter $\alpha \in \mathbb{R}$. We also prove that if $\alpha = 1/2$ the corresponding diffusion map with the Mahalanobis kernel approximates exactly the generator for SDE (1). Our proofs involve only elementary tools such as multivariable calculus and linear algebra. We will refer to the resulting diffusion map algorithm with the Mahalanobis kernel as `mmap`. We will compare its results to the original diffusion map algorithm with isotropic Gaussian kernel and refer to it as `dmap`.
- We apply `mmap` with $\alpha = 1/2$ to two common test systems: the alanine dipeptide molecule [7,13,19,26–28] and the Lennard-Jones cluster of 7 particles in 2 dimensions (LJ7) [29–31]. For both systems, we compute the committor function on trajectory data and provide validation for the results. For alanine dipeptide, we compare the committor obtained using `mmap` with the one computed using a finite difference method and demonstrate good agreement between these two. We contrast the committor obtained from `mmap` with the one obtained by `dmap` and show that the latter is notably less accurate. We also compute the reactive currents for both `mmap` and `dmap` and obtain an estimate for the transition rate. In addition, we investigate the dependence of the error in the estimate for the committor on the scaling parameter ϵ in the kernel of `mmap` and `dmap` and conclude that `mmap` is consistently more accurate and at least as robust as `dmap`. For LJ7, we conduct a committor analysis, a simulation-based validation technique for the committor [5,32,33]. Our results indicate that `mmap` produces a reasonable approximation for

the 1/2-isocommittor surface, while `dmap` places this surface at an utterly wrong place in the collective variable space.

The rest of the paper is organized as follows. In Section 2 we review relevant concepts from MD in collective variables, transition path theory and diffusion maps. Our theoretical results are presented in Section 3. Applications to alanine dipeptide and LJ7 are detailed in Section 4. Concluding remarks are given in Section 5. Various technical points and proofs are worked out in the appendices.

2. Background

In this section, we give a quick overview on collective variables, transition path theory, and diffusion maps.

2.1. Effective dynamics and collective variables

Our primary interest in this work is in datasets arising in MD simulations. We consider the *overdamped Langevin equation*, a simplified model for molecular motion which describes the molecular configuration in terms of the positions y of its atoms:

$$dy_t = -\nabla V(y_t)dt + \sqrt{2\beta^{-1}}dw_t, \tag{2}$$

where $y \in \mathbb{R}^m$, $V : \mathbb{R}^m \to \mathbb{R}$ is a potential function, $\beta^{-1} = k_bT$ is temperature in units of Boltzmann's constant, t is time, and w_t is a Brownian motion in $\mathbb{R}^m$. Given certain conditions on the potential V, a system governed by overdamped Langevin dynamics is ergodic with respect to the *Gibbs distribution* $\mu(y) = Z_V^{-1}e^{-\beta V(y)}$, where Z_V is a normalizing constant. The overdamped Langevin dynamics has infinitesimal generator

$$\mathcal{L}f = \beta^{-1}\Delta f - \nabla f \cdot \nabla V = \beta^{-1}e^{\beta V}\nabla \cdot (e^{-\beta V}\nabla f) \tag{3}$$

defined for twice continuously differentiable, square-integrable functions f.

As mentioned in the introduction, the number of atoms in biomolecules is typically very large. Even for such a small molecule as alanine dipeptide, the number of atoms is 22 and results in a 66-dimensional configuration space ($m = 66$). Furthermore, to describe the state of a biomolecule one does not need atomic positions y per se but rather certain functions in y specifying desired geometric characteristics. Therefore, to reduce the dimensionality and obtain a more useful and comprehensive description of the system-at-hand, one uses *collective variables* (CVs). CVs are functions of the atomic coordinates designed to give a coarse-grained description of the system's dynamics, preserving transitions between metastable states but erasing small-scale vibrations. Physical intuition has traditionally driven the choice of collective variables including dihedral angles, intermolecular distances, macromolecular distances and various experimental measurements.

We denote the set of CVs as the vector-valued function $x = \theta(y)$. Since our goal is to compute the committor (its precise definition is given in Section 2.2), a chosen set of CVs is good if the committor is well-approximated by a function that depends only on $\theta(y)$. The dynamics in collective variables is approximated by a diffusion process governed by [5] SDE (1):

$$dx_t = \left[-M(x_t)\nabla F(x_t) + \beta^{-1}\nabla \cdot M(x_t)\right]dt + \sqrt{2\beta^{-1}}M(x_t)^{1/2}dw_t.$$

Here, $F(x)$ is the *free energy* and $M(x)$ is the *diffusion matrix*, a symmetric positive definite matrix function. They are computed by averaging the appropriate functions of $x = \theta(y)$ over all $y \in \mathbb{R}^m$ with respect to

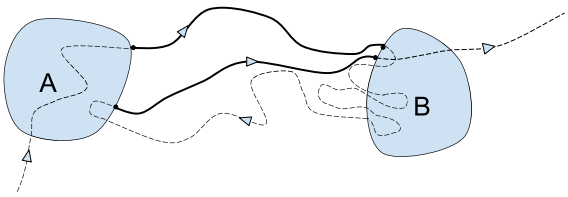

Fig. 1. A segment of a long trajectory. Reactive pieces from reactant state A to product state B are shown with solid lines.

the invariant density $\mu(y) = Z_V^{-1} e^{-\beta V(y)}$. The exact formulas for $F(x)$ and $M(x)$ and their evaluation in practice are detailed in Appendix A.

The generator for SDE (1) is given by

$$\mathcal{L}f = (-M\nabla F + \beta^{-1}(\nabla \cdot M))^\top \nabla f + \beta^{-1}\mathsf{tr}[M\nabla\nabla f], \tag{4}$$

which can also be written in divergence form as

$$\mathcal{L}f = \beta^{-1} e^{\beta F} \nabla \cdot (e^{-\beta F} M \nabla f). \tag{5}$$

One can check [2] that the process given by an SDE of the form (1) is reversible and the invariant probability measure for (1) is $\rho(x) = Z^{-1} e^{-\beta F(x)}$, the Gibbs measure. Moreover, we would like to remark that any reversible diffusion process must be of the form (1) [34].

2.2. Transition path theory in collective variables

Throughout this section we assume that the system under consideration is governed by the overdamped Langevin SDE in collective variables (1). Suppose we have an infinite trajectory $\{x_t\}_{t=0}^{\infty}$. Further, suppose that we have designated a priori two minima x_A, x_B of the potential F with corresponding disjoint open subsets $A \ni x_A, B \ni x_B$ which we refer to as the *reactant* and *product* sets respectively. Transition path theory (TPT) [2,3] is a mathematical framework to analyze statistics of transitions between the reactant A and the product B. The subject of TPT is the ensemble of *reactive trajectories*, defined as any continuous pieces of the trajectory x_t which start at ∂A and end at ∂B without returning to ∂A in-between (see Fig. 1). Key concepts of TPT are the *forward and backward committor functions* with respect to A and B. Since the governing SDE (1) is reversible, the forward q_+ and backward q_- committors are related via $q_- = 1 - q_+$ [2]. Hence, for brevity, we will refer to the forward committor as *the committor* and denote it merely by $q(x)$. The committor q has a straightforward probabilistic interpretation:

$$q(x) = \mathbb{P}(\tau_B < \tau_A \mid x_0 = x), \tag{6}$$

where $\tau_A := \inf\{t > 0 \mid x_t \in A\}$ and $\tau_B = \inf\{t > 0 \mid x_t \in B\}$ are the *first entrance times* of the sets A and B, respectively. In words, $q(x)$ is the probability that a trajectory starting at x will arrive at the product set B before arriving at the reactant set A. One can show that q satisfies the boundary value problem [2]

$$\begin{cases} \mathcal{L}q(x) = 0 & x \notin (A \cup B) \\ q(x) = 0 & x \in \partial A \\ q(x) = 1 & x \in \partial B, \end{cases} \tag{7}$$

where $\mathcal{L}$ is the infinitesimal generator (4). Once the committor is computed, one can find the reactive current that reveals the mechanism of the transition process:

$$\mathcal{J} = \beta^{-1} Z^{-1} e^{-\beta F(x)} M(x) \nabla q(x). \tag{8}$$

The integral of the flux of the reactive current through any hypersurface Σ separating the sets A and B gives the reaction rate:

$$\nu_{AB} := \lim_{t\to\infty} \frac{N_{AB}}{t} = \int_{\Sigma} \mathcal{J} \cdot \hat{n} d\sigma, \tag{9}$$

where N_{AB} is the total number of transitions from A to B performed by the system within the time interval $[0, t]$, and $\hat{n}$ is the unit normal to the surface Σ pointing in the direction of B.

Transition path theory has been extended to Markov jump processes [35,36] on a finite state space $\mathcal{S}$, $|\mathcal{S}| = n$, defined by the generator matrix L satisfying

$$\begin{cases} \sum_{j\in\mathcal{S}} L_{ij} = 0, & i \in \mathcal{S} \\ L_{ij} \geq 0, & i \neq j,\ i, j \in \mathcal{S}. \end{cases} \tag{10}$$

The settings and concepts in discrete TPT mimic those from its continuous counterpart. In particular, the committor is the vector $[q] = [q_1, \ldots, q_n]^\top$ with

$$[q]_i = \mathbb{P}(\tau_B < \tau_A \mid X_0 = i).$$

Analogously, $[q]$ solves the matrix equation

$$\begin{cases} [Lq]_i = 0 & i \in \mathcal{S}\backslash(A \cup B), \\ [q]_i = 0 & i \in A, \\ [q]_i = 1 & i \in B. \end{cases} \tag{11}$$

Our goal is the following. Let $\{x_i\}_{i=1}^n$ be a dataset sampled from SDE (1). We need to construct a *discrete* generator L such that, given a smooth scalar function $f(x)$, we have

$$\sum_{j=1}^{n} L_{ij} f(x_j) \approx \mathcal{L} f(x_i) \quad \text{for each} \quad x_i,\ 1 \leq i \leq n,$$

where $\mathcal{L}$ is the generator of (1) defined in (4). We refer to this approximation as a *pointwise approximation of the generator* with respect to the dataset. Given a pointwise approximate generator matrix L, we can then pointwise approximate the continuous committor q via our solution to the discrete committor equation (11). In this work, we will show that the Mahalanobis diffusion map (`mmap`) yields the desired approximation.

2.3. Diffusion maps

The diffusion map algorithm takes as input a dataset $X = \{x_i\}_{i=1}^n \subset \mathbb{R}^d$ of independent samples drawn from a distribution $\rho(x)$ that does not need to be known in advance. The manifold learning framework assumes that X lies near a manifold $\mathcal{M}$ which has low intrinsic dimension. Pairwise similarity of data is encoded through a kernel function $k_\epsilon(x, y)$ whose simplest form is given by

$$k_\epsilon(x, y) = \exp\left(-\frac{||x - y||^2}{2\epsilon}\right). \tag{12}$$

The user-chosen parameter $\epsilon > 0$ is the *kernel bandwidth parameter* (or the *scaling parameter*). The original diffusion map algorithm [15] requires an isotropic kernel $h_\epsilon(||x-y||^2)$ with exponential decay as $\|x-y\| \to \infty$. Let us describe the construction of a diffusion map following the steps in [15]. The kernel $k_\epsilon(x,y)$ is used to define an $n \times n$ similarity matrix K_ϵ with $[K_\epsilon]_{ij} = k_\epsilon(x_i, x_j)$. The strong law of large numbers implies that for a scalar $f(x)$ on $\mathbb{R}^d$ we have:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} k_\epsilon(x_i, x_j) f(x_j) = \int_{\mathbb{R}^d} k_\epsilon(x_i, y) f(y) \rho(y) dy \quad \text{almost surely.} \tag{13}$$

It follows from the Central Limit Theorem that the error of this estimate decays as $\mathcal{O}(n^{-\frac{1}{2}})$. The action of the kernel on sufficiently large datasets is approximated by an integral operator $\mathcal{G}_\epsilon$ defined by

$$(\mathcal{G}_\epsilon f)(x) := \int_{\mathbb{R}^d} k_\epsilon(x, y) f(y) dy. \tag{14}$$

Namely, for a sufficiently large dataset,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} k_\epsilon(x_i, x_j) f(x_j) = \mathcal{G}_\epsilon(f\rho)(x_i) \text{ almost surely.} \tag{15}$$

The main innovation of the diffusion map algorithm [15] in comparison with Laplacian eigenmap [18] is the introduction of the parameter $\alpha \in \mathbb{R}$ allowing us to control the influence of the density ρ and approximate a whole family of differential operators. Let us review the construction of diffusion maps. The first step is to compute the normalizing factor $\rho_\epsilon(x)$, where

$$\rho_\epsilon(x) = \int_{\mathbb{R}^d} k_\epsilon(x, y) \rho(y) dy. \tag{16}$$

The *right-normalized* kernel is defined as

$$k_{\epsilon,\alpha}(x, y) = \frac{k_\epsilon(x, y)}{\rho_\epsilon^\alpha(y)}. \tag{17}$$

We note that one can write the action of the right-normalized kernel on density-weighted functions $f(x)\rho(x)$ as

$$\int_{\mathbb{R}^d} k_{\epsilon,\alpha}(x, y) f(y) \rho(y) dy = \int_{\mathbb{R}^d} k_\epsilon(x, y) f(y) \rho(y) \rho_\epsilon^{-\alpha}(y) dy.$$

Hence the right normalization regulates the influence of the density $\rho(x)$ in Monte Carlo integrals. We define vector p_ϵ as the vector of row sums of the matrix K_ϵ:

$$[p_\epsilon]_i = \sum_{j=1}^{n} [K_\epsilon]_{i,j}. \tag{18}$$

We define the diagonal matrix $D_\epsilon = \mathsf{diag}(p_\epsilon)$. Then, the discrete counterpart for the *right-normalized* kernel (17) is

$$K_{\epsilon,\alpha} := K_\epsilon D_\epsilon^{-\alpha}. \tag{19}$$

Next, we fix

$$\rho_{\epsilon,\alpha}(x) := \int_{\mathbb{R}^d} k_{\epsilon,\alpha}(x,y)\rho(y)dy \tag{20}$$

to *left-normalize* the kernel, and define the Markov operator $\mathcal{P}_{\epsilon,\alpha}$ on f as

$$\mathcal{P}_{\epsilon,\alpha}f(x) = \frac{\int_{\mathbb{R}^d} k_{\epsilon,\alpha}(x,y)f(y)\rho(y)dy}{\rho_{\epsilon,\alpha}(x)}. \tag{21}$$

Finally, we define a family of operators $\mathcal{L}_{\epsilon,\alpha}$ as

$$\mathcal{L}_{\epsilon,\alpha}f = \frac{\mathcal{P}_{\epsilon,\alpha}f - f}{\epsilon} \equiv \frac{1}{\epsilon}\left(\frac{\mathcal{G}_\epsilon(f\rho\rho_\epsilon^{-\alpha})}{\mathcal{G}_\epsilon(\rho\rho_\epsilon^{-\alpha})} - f\right). \tag{22}$$

To obtain its discrete counterpart, we form a diagonal matrix $D_{\epsilon,\alpha}$ with row sums of $K_{\epsilon,\alpha}$ along its diagonal and use it to *left-normalize* the matrix $K_{\epsilon,\alpha}$ and get the Markov matrix $P_{\epsilon,\alpha} = D_{\epsilon,\alpha}^{-1}K_{\epsilon,\alpha}$. Then the family of *discrete operators* $L_{\epsilon,\alpha}$ is defined as

$$L_{\epsilon,\alpha} := \frac{P_{\epsilon,\alpha} - I}{\epsilon}. \tag{23}$$

The discrete and continuous operators $L_{\epsilon,\alpha}$ and $\mathcal{L}_{\epsilon,\alpha}$ relate via the pointwise infinite data limit. Let us fix an arbitrary point x_1 drawn from the invariant density ρ and keep adding more points drawn from $\rho(x)$ into a dataset containing x_1. Then

$$\lim_{n\to\infty} [L_{\epsilon,\alpha}[f]]_1 = [\mathcal{L}_{\epsilon,\alpha}]\, f(x_1), \quad \text{almost surely} \tag{24}$$

with error decaying as $O(n^{-\frac{1}{2}})$. It is proven in [15] that[1]

$$\lim_{\epsilon\to 0} \mathcal{L}_{\epsilon,\alpha}f = \frac{1}{2}\left(\frac{\Delta(\rho^{1-\alpha}f) - f\Delta(\rho^{1-\alpha})}{\rho^{1-\alpha}}\right). \tag{25}$$

If the input dataset X comes from the overdamped Langevin SDE (2), and hence the invariant measure is Gibbs, i.e., $\rho(x) = Z^{-1}e^{-\beta V(x)}$, and the kernel is given by (12), then

$$\lim_{\epsilon\to 0} \mathcal{L}_{\epsilon,\alpha}f = \frac{1}{2}\left[\Delta f - 2\beta(1-\alpha)\nabla f^\top \nabla V\right]. \tag{26}$$

The series of steps described here leading to the construction of the matrix operator $L_{\epsilon,\alpha}$ (23) will be referred to as `dmap` (an abbreviation for the diffusion map algorithm).

The renormalization parameter α tunes the influence of the density ρ in the operator $\mathcal{L}_{\epsilon,\alpha}$. Setting $\alpha = 0$ yields, up to a multiplicative constant $1/2$, the standard graph Laplacian, which converges to the Laplace-Beltrami operator only if ρ is uniform. With $\alpha = 1$ the density $\rho(x)$ is reweighted so that the limiting density for $\epsilon \to 0, n \to \infty$ is uniform and the Laplace-Beltrami operator $\mathcal{L}f = \frac{1}{2}\Delta f$ is recovered. Setting $\alpha = 1/2$ recovers the backward Kolmogorov operator,

$$\lim_{\epsilon\to 0} \mathcal{L}_{\epsilon,1/2}f = \frac{\beta}{2}\left[\beta^{-1}\Delta f - \nabla f^\top \nabla V\right] \equiv \frac{\beta}{2}\mathcal{L}f, \tag{27}$$

which is needed for computing the committor.

[1] We believe that a multiplicative constant is missing in Theorem 2 and in Proposition 10 in [15].

The diffusion map algorithm has also seen many other modifications and improvements which are used heavily in practice. A primary development concerns the kernel bandwidth parameter ϵ, which usually requires extensive tuning in practice. Diffusion map variations such as locally scaled diffusion maps [7] and variable bandwidth diffusion kernels [37] utilize a bandwidth which varies at each data point and can improve stability as well as accuracy at the boundary points of a dataset.

2.4. Diffusion maps for data coming from SDEs with multiplicative noise

An important limitation of the original class of diffusion maps with isotropic kernels [15] is that it can only approximate infinitesimal generators of the form (26) that are relevant only for gradient flows [38,39]. This limitation is caused by the fact that the construction relies on the sampling density of the data but not dynamical properties of the data. For example, the reversible diffusion process in collective variables (1) has the Gibbs distribution $\rho(x)$ and diffusion matrix $M(x)$, but application of diffusion maps will approximate the generator of gradient dynamics with density $\rho(x)$ and *constant* diffusion matrix [38]. Hence, to approximate generators for diffusion processes with multiplicative noise, a modified diffusion maps algorithm is required [24,38].

2.4.1. Mahalanobis diffusion maps

The foundational approach for applying diffusion maps to diffusion processes with multiplicative noise is proposed by Singer and Coifman (2008) [24]. They consider a diffusion process

$$dz_t = b(z_t)dt + \sqrt{2}dw_t, \quad z_t \in \mathcal{M}, \tag{28}$$

where $\mathcal{M}$ is a d-dimensional manifold. The generator for this process is given by

$$\mathcal{L}_z = \Delta + b \cdot \nabla. \tag{29}$$

The dynamics (28) are considered as the unobserved intrinsic dynamics of interest, while the observed dynamics is the process $x_t = \zeta(z_t)$, where ζ is an injective, smooth function from $\mathcal{M}$ to $\mathbb{R}^m$, where $m \geq d$. To elucidate the key idea from [24], we assume that $\mathcal{M} \equiv \mathbb{R}^d$, $m = d$, and that the mapping $x = \zeta(z)$ is a diffeomorphism. From Ito's Lemma, the differential of ζ is

$$d\zeta_i(z_t) = \sum_{j=1}^{d} \left(\frac{\partial \zeta_i(z_t)}{\partial z_j} b_j(z_t) + \frac{\partial^2 \zeta_i(z_t)}{\partial z_j^2} \right) dt + \sqrt{2} \sum_{j=1}^{d} \frac{\partial \zeta_i(z_t)}{\partial z_j} [dw_t]_j. \tag{30}$$

Thus, the noise term for $d\zeta(z_t)$ is given by $\sqrt{2}J(z_t)dw_t$, where $J(z_t)$ is the Jacobian of ζ with entries $[J(z)]_{ij} = \frac{\partial \zeta_i}{\partial z_j}$.

The crucial fact utilized in [24] is the following relationship between the $(JJ^\top)^{-1}$-weighted quadratic form in the space of observed variables $x = \zeta(z)$ and the Euclidean distance in the z-space:

$$\begin{aligned} &\frac{1}{2}(\zeta(z) - \zeta(z'))^\top \left[(JJ^\top)^{-1}(x) + (JJ^\top)^{-1}(y) \right] (\zeta(z) - \zeta(z')) \\ &\quad = ||z - z'||^2 + \mathcal{O}(||z - z'||^4). \end{aligned} \tag{31}$$

This relationship motivated the introduction of the anisotropic kernel

$$k_\epsilon(x, y) = \exp\Big(-\frac{1}{4\epsilon}(x - y)^\top ((JJ^\top)^{-1}(x) + (JJ^\top)^{-1}(y))(x - y) \Big). \tag{32}$$

Algorithm 1: Mahalanobis Diffusion Map Algorithm (`mmap`).

Input: data $X = \{x_i\}_{i=1}^n$, diffusion matrices $\{M(x_i)\}_{i=1}^n$, bandwidth ϵ, renormalization parameter α
Output: Matrix operator $L_{\epsilon,\alpha}$
Construct kernel using (33)
$[K_\epsilon]_{i,j} = k_\epsilon(x_i, x_j), \quad i,j = 1,\dots,n$
Find row sums of the kernel matrix and form a diagonal matrix
$[p_\epsilon]_i = \sum_{j=1}^n [K_\epsilon]_{ij}, \quad i = 1,\dots,n$
$D_\epsilon = \mathsf{diag}\{[p_\epsilon]_1,\dots,[p_\epsilon]_n\}$
Right normalize the kernel
$[K_{\epsilon,\alpha}] := K_\epsilon D_\epsilon^{-\alpha}$
Left normalize the kernel
$[p_{\epsilon,\alpha}]_i = \sum_{j=1}^n [K_{\epsilon,\alpha}]_{ij}, \quad i = 1,\dots,n$
$D_{\epsilon,\alpha} = \mathsf{diag}\{[p_{\epsilon,\alpha}]_1,\dots,[p_{\epsilon,\alpha}]_n\}$
$P_{\epsilon,\alpha} := D_{\epsilon,\alpha}^{-1} K_{\epsilon,\alpha}$
Construct generator
$L_{\epsilon,\alpha} = \dfrac{P_{\epsilon,\alpha} - I}{\epsilon}$

The diffusion map with kernel (33) approximates the generator $\mathcal{L}_z$ for the unknown latent dynamics z_t in the case where $b(z_t) = -\nabla U(z_t)$ for some potential $U(z)$. The fact that the diffusion matrix is of the form $JJ^\top$ is essential for the proof of this approximation presented in [24]. Relationship (31) essentially reduces this to the proof for diffusion maps with rotationally symmetric Gaussian kernel in [15].

The algorithm proposed in [24] and variants have been applied to multiscale fast slow-processes [24,40], nonlinear filtering problems [41], optimal transport and data fusion problems [42,43], chemical reaction networks [44], localization in sensor networks [45], and molecular dynamics [46]. Further, kernel (33) has recently been used for isometric embeddings to high-dimensional latent spaces [47] and for deep learning frameworks [45,47].

This work addresses the case where diffusion matrix $M(x)$ in SDE (1) *is not necessarily decomposable as* $M(x) = \left(JJ^\top\right)(x)$ and hence there is no relationship of the form (31) to utilize. Following [24], we use the Mahalanobis kernel

$$k_\epsilon(x,y) = \exp\Big(-\frac{1}{4\epsilon}(x-y)^\top (M^{-1}(x) + M^{-1}(y))(x-y)\Big). \tag{33}$$

Other than the choice of the kernel, the Mahalanobis diffusion map algorithm (`mmap`) follows the steps of the diffusion map algorithm (`dmap`) detailed in Section 2.3. For the reader's convenience, we summarize `mmap` in Algorithm 1. The family of differential operators approximated by `mmap` will be derived in Section 3.

Remark 2.1. The term *Mahalanobis kernel* is related to the Mahalanobis distance. If data points x and y are sampled from a multivariate Gaussian distribution with covariance matrix C, then

$$(x-y)^\top C^{-1}(x-y)$$

is the squared Mahalanobis distance between x and y. In the orthonormal basis of eigenvectors of C, the difference between each component of x and y is normalized by the corresponding variance, which reflects the difficulty to deviate along each direction. Hence, kernel (33) is a decaying exponential function of an approximate squared Mahalanobis distance. Therefore, it is designed to account for anisotropy of the diffusion process the data is coming from.

2.4.2. Local kernels

A further development facilitating data-driven analysis of anisotropic diffusion processes was done by Berry and Sauer (2016) [39]. Their local kernels theory generalizes theoretical results of [15,24] to a class

of anisotropic kernels which utilize user-defined drift vectors $b(x_i)$ and diffusion matrices $A(x_i)$. The local kernel approach has been extended to related work in solving elliptic PDEs with diffusion maps [48] and to computing reaction coordinates for molecular simulation [38]. In [38] the authors incorporate arbitrary sampling densities into the local kernel approach [39] and prove that for user-defined drift vector $b(x)$ and user-defined diffusion matrix $A(x)$, kernels of the form

$$k_\epsilon^{A,b}(x,y) = \exp\left(-\frac{1}{4\epsilon}(x-y+\epsilon b(x))^\top A^{-1}(x)(x-y+\epsilon b(x))\right), \tag{34}$$

applied to data $\{x_i\}_{i=1}^n$ with arbitrary density can be normalized (similarly to the diffusion map with $\alpha = 1$) to approximate the differential operator

$$\mathcal{L}f(x) = b(x)\nabla f(x) + \mathsf{tr}[A(x)\nabla\nabla f(x)]. \tag{35}$$

Equation (35) describes a broader class of generators than (4), which is advantageous. On the downside, the local kernel diffusion map algorithm of [38] requires drift estimates at all data points, as well as a second kernel $k_{\tilde\epsilon}(x,y)$ with additional scaling parameter $\tilde\epsilon$ in order to normalize the density of the dataset. As a result, implementation of the local kernel approach requires adjustment of two scaling parameters ϵ and $\tilde\epsilon$, which can be challenging.

We are primarily interested in the reversible dynamics in collective variables coming from MD simulations. On the other hand, the density of the data is far from being uniform, and may change by orders of magnitude thereby complicating the tuning of scaling parameters. Therefore, we choose to use `mmap` rather than the local kernel approach.

3. Theoretical results

Our goal is to prove that the `mmap` algorithm (Algorithm 1) with $\alpha = 1/2$ approximates the generator (4) for SDE (1), the overdamped Langevin dynamics in collective variables. First we show that not every symmetric positive definite smooth matrix function $M(x)$ admits the decomposition $JJ^\top(x)$ where $J(x)$ is the Jacobian matrix function for some smooth vector-function. This will justify the lengthy calculation conducted in our proof of the main theorem (Theorem 3.3 below). Next, we derive the family of differential operators approximated by `mmap` with an arbitrary $\alpha \in \mathbb{R}$ (Theorem 3.3). Finally, we evaluate the resulting differential operator at $\alpha = 1/2$ and show that it is the generator for (1) (Corollary 3.1).

3.1. Not every diffusion matrix is associated with a variable change

The fact that not every smooth symmetric positive definite matrix function $M(x)$ can be decomposed as

$$M(x) = J(x)J(x)^\top \quad \text{where} \quad J = \left(\frac{\partial f_i}{\partial x_j}\right)_{i,j=1}^d \tag{36}$$

for some smooth vector-function $f : \Omega \to \mathbb{R}^d$, where Ω is an open set in $\mathbb{R}^d$, is not new but it is not widely known. The non-existence of decomposition (36) is pointed out for the position-dependent diffusion matrix in the Moro-Cardin 2D example [49] by M. Johnson and G. Hummer [50] with a reference to the textbook by H. Risken [25] (Section 4.10), where the general criterion for the existence of decomposition (36) consisting in vanishing a certain complicated differential form is presented.

Here we will give a simple proof of the fact that not every symmetric positive definite smooth matrix function admits decomposition (36) by establishing a necessary condition for a class of 2×2 symmetric positive definite matrix functions

$$M(x,y) = m^2(x,y) I_{2\times 2}, \quad m(x,y) \in C^2(\Omega), \quad m(x,y) > 0 \ \forall (x,y) \in \Omega, \tag{37}$$

for decomposition (36) to exist. Precisely, the necessary condition requires the function $\log m(x,y)$ to be *harmonic*. Moreover, if the open set Ω is simply connected, this condition is also sufficient for the class (37).

Theorem 3.1. *Let $M(x,y)$ be a symmetric positive definite matrix function of the form (37) where $m(x,y)$ is a positive twice continuously differentiable function in an open set $\Omega \subset \mathbb{R}^2$ and $I_{2\times 2}$ is a 2×2 identity matrix. Suppose that M admits decomposition $M(x,y) = JJ^\top(x,y)$ where $J(x,y)$ is the Jacobian matrix of some twice continuously differentiable vector-function $f : \Omega \to \mathbb{R}^2$. Then $\log m(x,y)$ must be harmonic, i.e.,*

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right) \log m(x,y) = 0 \quad \forall (x,y) \in \Omega. \tag{38}$$

Moreover, if the open set Ω is simply connected, then (38) is also a sufficient condition for the existence of decomposition (36). If Ω is not simply connected, (38) is not a sufficient condition.

A proof of Theorem 3.1 is given in Appendix B.

Thus, any matrix function of the form (37) where $\log m(x,y)$ is not harmonic in Ω does not admit decomposition (36) in Ω. For example, the function m in the Moro-Cardin example [49]

$$m(x,y) = \left(1 + e^{-\frac{x^2+y^2}{2}}\right)^{-1/2} \tag{39}$$

is such that its logarithm is not harmonic:

$$\Delta \log m(r) = \frac{1}{2}\left[\frac{(2-r^2)(1+e^{-r^2/2})e^{-r^2/2} + r^2 e^{-r^2}}{(1+e^{-r^2/2})^2}\right], \quad r = \sqrt{x^2+y^2}.$$

Furthermore, any $d \times d$ twice continuously differentiable matrix function $M(x,y)$ that has a principal 2×2 submatrix of the form (37) where $\log m(x,y)$ is not harmonic in Ω does not admit decomposition (36).

3.2. The family of differential operators approximated by `mmap`

We will adopt three technical assumptions. The first one deals with the space of collective variables x:

Assumption 1. The range of x representing the set of collective variables constitutes a d-dimensional manifold $\mathcal{M}$ which is either $\mathbb{R}^d$, or the d-dimensional torus $\mathbb{T}^d$, or a direct product of torus $\mathbb{T}^k$ and $\mathbb{R}^{d-k}$. In all cases, $\mathcal{M}$ is of the form

$$\mathcal{M} = \mathbb{T}^k \times \mathbb{R}^{d-k}, \quad \text{for some} \quad 0 \le k \le d. \tag{40}$$

By the torus $\mathbb{T}^k$, $1 \le k \le d$, we mean the "flat" torus, i.e., the direct product of intervals with periodic boundary conditions, i.e.,

$$\mathbb{T}^k \times \mathbb{R}^{d-k} = [a_1, b_1] \times [a_2, b_2] \times \ldots \times [a_k, b_k] \times \mathbb{R}^{d-k},$$
$$(x_1, \ldots, x_{l-1}, a_l, x_{l+1}, \ldots, x_d) = (x_1, \ldots, x_{l-1}, b_l, x_{l+1}, \ldots, x_d), \quad 1 \le l \le k.$$

The metric on such a torus is locally Euclidean [51], i.e., within any open ball of radius

$$R_{Euc} := \min_{1\le l\le k} \frac{|b_l - a_l|}{2}. \tag{41}$$

Therefore, the metric on $\mathcal{M}$ is Euclidean if $\mathcal{M} = \mathbb{R}^d$ or locally Euclidean within any ball of radius R_{Euc} if $\mathcal{M} = \mathbb{T}^k \times \mathbb{R}^{d-k}$ for some $1 \leq k \leq d$.

Assumption 1 is nonrestrictive in view of chemical physics applications, as usually collective variables are dihedral angles or distances between certain atoms. For example, the alanine dipeptide molecule is represented in two or four dihedral angles, $\mathcal{M} = \mathbb{T}_2$ or $\mathbb{T}_4$, a 2D or a 4D torus respectively. Assumption 1 allows us to prove our main theoretical result from scratch using only elementary tools.

The second and third assumptions impose integrability and differentiability conditions on the diffusion matrix $M(x)$ and a class of functions $f : \mathcal{M} \to \mathbb{R}$ to which we apply the constructed family of operators. We need the following definition:

Definition 3.2. We say that a continuous function $f : \mathbb{R}^d \to \mathbb{R}$ grows not faster than a polynomial as $\|x\| \to \infty$ if there exist constants $A \geq 0$, $B \geq 0$, and $l \in \mathbb{N}$ such that

$$|f(x)| \leq A + B\|x\|^l \quad \forall x \in \mathbb{R}^d.$$

Assumption 2. The diffusion matrix $M(x)$ is symmetric positive definite. Its inverse $M^{-1}(x)$ is a four-times continuously differentiable matrix-valued function $M^{-1} : \mathcal{M} \to \mathbb{R}^{d\times d}$ and the determinant of $M^{-1}(x)$ is bounded away from zero. If the manifold $\mathcal{M}$ is unbounded (i.e., $0 \leq k \leq d-1$ in (40)), then the entries $(M^{-1})_{ij}(x)$ and their first derivatives $\frac{\partial (M^{-1})_{ij}(x)}{\partial x_\ell}$ grow not faster than a polynomial as $\|x\| \to \infty$.

Assumption 3. The function $f(x)$ is four-times continuously differentiable. If $\mathcal{M}$ is unbounded then $f(x)$ grows not faster than a polynomial as $\|x\| \to \infty$.

Now we are ready to formulate our convergence results for `mmap`.

Theorem 3.3. *Suppose a manifold $\mathcal{M}$ and a diffusion matrix $M(x) : \mathcal{M} \to \mathbb{R}^{d\times d}$ satisfy Assumptions 1 and 2 respectively. Let $\alpha \in \mathbb{R}$ be fixed and the kernel $k_{\epsilon,\alpha}$ be the Mahalanobis kernel (33), and the operator $\mathcal{L}_{\epsilon,\alpha}$ be constructed according to (16), (17), (20), (21), and (22). Then for any function $f(x) : \mathcal{M} \to \mathbb{R}$ satisfying Assumption (3) we have*

$$\begin{aligned}\lim_{\epsilon\to 0} \mathcal{L}_{\epsilon,\alpha} f(x) = \frac{1}{2}\left(\frac{\mathsf{tr}\left(M\left[\nabla\nabla\left[\rho^{1-\alpha} f\right] - f\nabla\nabla\rho^{1-\alpha}\right]\right)}{\rho^{1-\alpha}}\right) \\ + \frac{\alpha}{2}\left(\frac{\mathsf{tr}\left(M\left[\nabla\left[\rho^{1-\alpha} f\right] - f\nabla\left[\rho^{1-\alpha}\right]\right]\frac{\nabla|M|^{-\top}}{|M|^{-1}}\right)}{\rho^{1-\alpha}}\right) \\ - \left(\frac{\left[\nabla(f\rho^{1-\alpha}) - f\nabla(\rho^{1-\alpha})\right]^\top \omega_1]}{\rho^{1-\alpha}}\right) \quad \forall x \in \mathcal{M}, \end{aligned} \tag{42}$$

where $|M|$ denotes the determinant of M, and $\omega_1(x)$ is a vector-valued function defined by

$$\omega_{1,i}(x) := \frac{|M(x)|^{-1/2}}{(2\pi\epsilon)^{d/2}} \frac{1}{4\epsilon^2} \int_{\mathcal{M}} e^{-\frac{z^\top M(x)^{-1} z}{2\epsilon}} z_i \left[z^\top [\nabla M^{-1}(x) z] z\right] dz. \tag{43}$$

A proof of this theorem is done by a direct calculation of limit (42). It is carried out from scratch and involves only elementary tools from linear algebra and multivariable calculus. It is found in Appendix C. We remark that, in turn, the corresponding discrete operator applied to $f(x)$ discretized to a point cloud

drawn from the invariant density $\rho(x)$ and with $L_{\epsilon,\alpha}[f]$ defined by (18), (19), (23) converges pointwise with probability one to $\mathcal{L}_{\epsilon,\alpha}f$ as the number of data points tends to infinity.

Equation (42) defines a family of differential operators parametrized by $\alpha \in \mathbb{R}$. Since our goal is to compute committors, we are primarily concerned with approximating the generator (4) for the overdamped Langevin SDE in collective variables (1). Setting $\alpha = 1/2$ approximates the generator that we need:

Corollary 3.1. *Let $\mathcal{M}$ and $M(x)$ be as in Theorem 3.3. Suppose that the invariant density $\rho(x)$ takes the form of the Gibbs distribution $\rho(x) = Z^{-1}e^{-\beta F(x)}$ for free energy F, temperature parameter β^{-1}, and normalizing constant $Z = \int_{\mathcal{M}} e^{-\beta F(x)}dx$. Then for $\alpha = 1/2$ the limit (42) reduces to*

$$\lim_{\epsilon\to 0} \mathcal{L}_{\epsilon,1/2}f(x) = \frac{\beta}{2}\mathcal{L}f(x) \quad \forall x \in \mathcal{M}, \tag{44}$$

where

$$\mathcal{L}f = \left(-M\nabla F + \beta^{-1}\left(\nabla \cdot M\right)\right)^{\top} \nabla f + \beta^{-1}\mathsf{tr}[M\nabla\nabla f] \tag{45}$$

is the generator for the SDE

$$dx_t = \left[-M(x_t)\nabla F(x_t) + \beta^{-1}\nabla \cdot M(x_t)\right]dt + \sqrt{2\beta^{-1}}M^{1/2}(x_t)dw_t. \tag{46}$$

A proof of Corollary 3.1 is found in Appendix D. Our main interest is in solving the committor PDE (7). Our approach consists in approximating the generator $\mathcal{L}$ in (7) by the matrix operator $L_{\epsilon,1/2}$ which converges to $\mathcal{L}_{\epsilon,1/2}$ as the number of data points n tends to infinity as $O(n^{-1/2})$.

Finally, we remark that the use of the symmetric Mahalanobis kernel (33) is essential for the convergence of $\mathcal{L}_{\epsilon,1/2}f(x)$ to the generator (45). If one processes data sampled from a long trajectory of SDE (46) with `dmap`, i.e., implements the isotropic Gaussian kernel (12) in the diffusion map algorithm, one obtains an approximation to the generator for the diffusion process governed by

$$dx_t = -\nabla F(x_t) + \sqrt{2\beta^{-1}}dw_t$$

which has the same invariant density $Z^{-1}\exp(-\beta F(x))$ as (46) but a different drift and a different diffusion matrix. If one replaces the half-sum $\frac{1}{2}(M(x) + M(y))$ in (33) with $M(x)$, all terms containing derivatives of M in (42) do not arise, and only the first term in (42) remains. For $\alpha = 1/2$ this yields $-M\nabla F \cdot \nabla f + \beta^{-1}\mathsf{tr}[M\nabla\nabla f]$, the generator for the dynamics

$$dx_t = -M(x_t)\nabla F(x_t)dt + \sqrt{2\beta^{-1}}M^{1/2}(x_t)dw_t,$$

which approximates (46) only if M is constant or β^{-1} is small. In our examples presented in the next section, M varies considerably and β^{-1} is not so small, rendering the term $\beta^{-1}\nabla \cdot M(x_t)$ non-negligible.

4. Examples

In this section, we test `mmap` on two examples: alanine dipeptide and Lennard-Jones-7 in 2D. The results obtained with `mmap` will be validated by comparing them to results of other established methods and contrasted to those of the diffusion map with isotropic Gaussian kernel (`dmap`). In view of the remark at the end of Section 3, the fact that the committors obtained using `dmap` are significantly less accurate than the `mmap` committors is unsurprising.

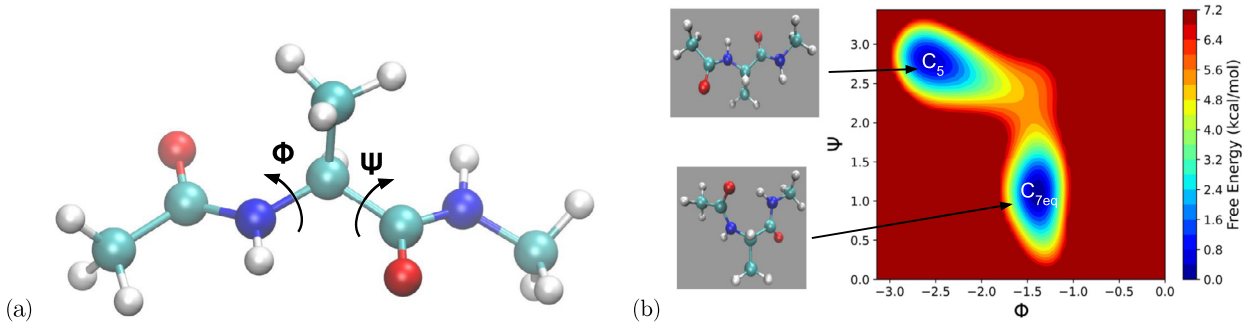

Fig. 2. (a): Structure of alanine dipeptide and dihedral angles Φ and Ψ serving as collective variables. (b): Free energy surface of alanine dipeptide in vacuum at temperature $T = 300$ K in vicinity of C5 and C7eq minima, in Φ, Ψ coordinates. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

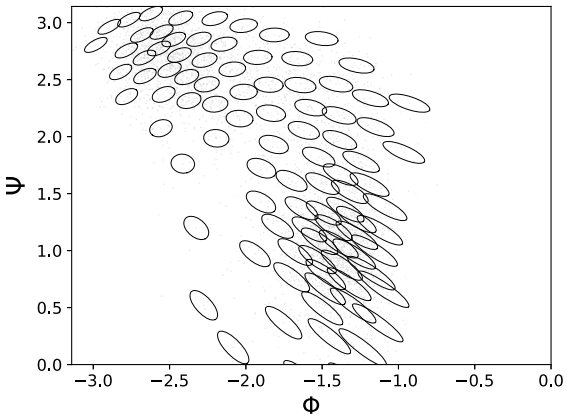

Fig. 3. Ellipses corresponding to the principal components of the estimated diffusion matrices for the alanine dipeptide data (faint gray dots). Each ellipse is plotted with center on the point whose diffusion matrix it represents. The ellipses are plotted on a representative subsampling of the trajectory data.

4.1. Transitions between metastable states C5 and C7eq in alanine dipeptide

Alanine dipeptide, a small biomolecule comprising 22 atoms, is a popular test example in chemical physics [5,4,52,13]. A typical set of collective variables effectively representing its motion consists of four or just two dihedral angles. We choose the set of only two dihedral angles Φ and Ψ shown in Fig. 2(a). Their range comprises a two-dimensional torus, i.e. the manifold $\mathcal{M}$ is $\mathbb{T}^2$.

4.1.1. Obtaining input data

We used a velocity-rescaling thermostat to set the temperature to 300 K in a vacuum and ran a 1 nanosecond trajectory under constant number, volume, and temperature (NVT) conditions, integrating Newton's equations of motion with timestep 2 femtoseconds using the molecular dynamics software GROMACS [53]. For use with diffusion maps, we subsampled the trajectory at equispaced intervals of timesteps to obtain $n = 5000$ data points $\{x_i\}_{i=1}^{5000}$ with $x_i \in \mathbb{T}^2$. The diffusion matrices $M(x_i)$ were obtained following the methodology of [5] (see Appendix A) and are visualized in Fig. 3. The reactant and product sets A and B are the small ellipses centered at the C5 and C7eq minima in the (Φ, Ψ)-space shown in Fig. 2(b). In Φ, Ψ coordinates the C5 and C7eq minima are $(-2.548, 2.744)$ and $(-1.419, 1.056)$ respectively, and the ellipses shown are the level sets of the free energy F at 1.4 kcal/mol.

To compare the committors computed via `mmap` and `dmap` with the one obtained by a traditional PDE solver, we discretized the range $[-\pi, \pi]^2$ of (Φ, Ψ) into a uniform square mesh 128×128 as in [4] and generated $M(x)$ and $\nabla F(x)$ using the procedure from [5], summarized in Appendix A. We then posed a

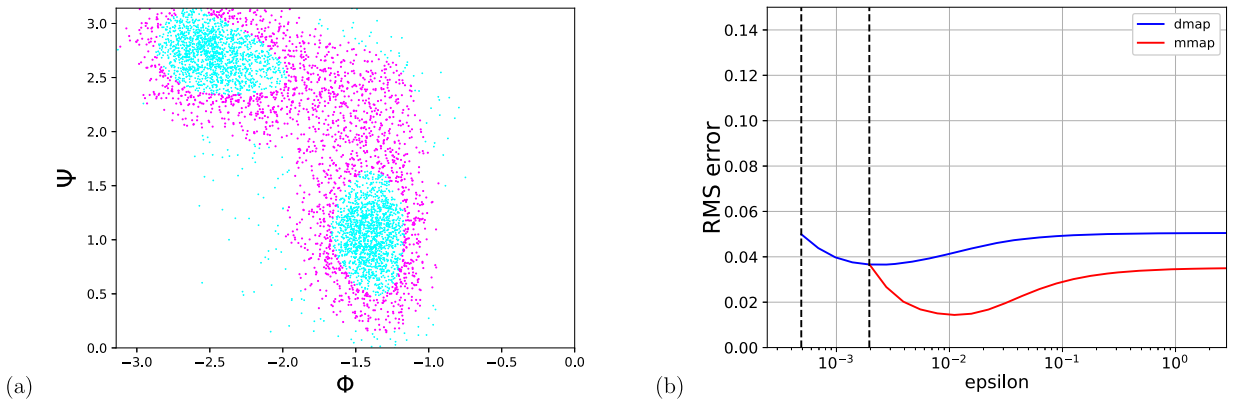

Fig. 4. (a): Alanine dipeptide dataset. Its subset lying in the region of interest marked with magenta is used for computing the RMS error. (b): RMS errors for `dmap` and `mmap` committors as functions of the scaling parameter ϵ. The dotted lines indicate lower bound for ϵ-values related to the minimal distance between data points.

boundary-value problem for the committor PDE from (4), (7) and solved it using a finite difference scheme with central second-order accurate approximations to the derivatives.

Often (see e.g. [5,11] and many other works) a much rarer transition in alanine dipeptide is studied: the one between the combined metastable state comprising C5 and C7eq and the metastable state called C7ax located near $\Phi = 75°$, $\Psi = -75°$. We chose the transition between C5 and C7eq for our tests because it can be easily sampled at room temperature $T = 300$ K. It is essential for `mmap` to have sufficient data coverage of the transition region, and the data must be sampled from the invariant distribution. The study of this transition gives us another benefit: unlike that for the transition between (C5, C7eq) and C7ax, the free energy barrier between C5 and C7eq is not large in comparison with k_bT. This renders the term $\beta^{-1}\nabla \cdot M(x)$ in SDE (1) non-negligible which is nonzero if and only if $M(x)$ is nonconstant. As a result, the contrast between the results of `mmap` and `dmap` is amplified. We leave the task of upgrading `mmap` to make it applicable to datasets obtained using enhanced sampling techniques for the future.

4.1.2. Results and validation

We computed the committor using the `mmap` and `dmap` algorithms with k-nearest neighbor (kNN) sparsified kernels and a large range of values of the scaling parameter ϵ. This range is naturally bounded from above and below by the diameter of the point cloud and by the minimal distance between data points, respectively. In addition, we computed the committor by solving the boundary-value problem for the committor PDE using finite differences as mentioned in Section 4.1.1 and took it as a ground truth q_{true}. To quantify the error of the `mmap` and `dmap` committors, we evaluated the root-mean-square (RMS) error

$$\text{RMS error} = \sqrt{\frac{\sum_{j=1}^{n}(q_{true}(x_j) - q_{approx}(x_j))^2}{n}},$$

where $\{x_j\}_{j=1}^n$ are the data points. The finite difference solution q_{true} was evaluated on the data through bilinear interpolation. It is clear that the committor computed with diffusion maps cannot be expected to be accurate on the outskirts of the dataset where the data coverage is insufficient. On the other hand, the `mmap` and `dmap` committors are exact at A and B by construction and highly accurate near them, and these are the regions containing the majority of data points as they are sampled from the invariant density. We care the most about the accuracy of the `mmap` and `dmap` committors in the transition region. Therefore, we select the subset of points marked with magenta dots in Fig. 4(a). The graphs of the RMS errors for `mmap` (red) and `dmap` (blue) over this subset as functions of ϵ are displayed in Fig. 4(b). The epsilon values minimizing the RMS error of the `mmap` and `dmap` committor are $\epsilon = 0.01$ and $\epsilon = 0.003$ with RMS errors

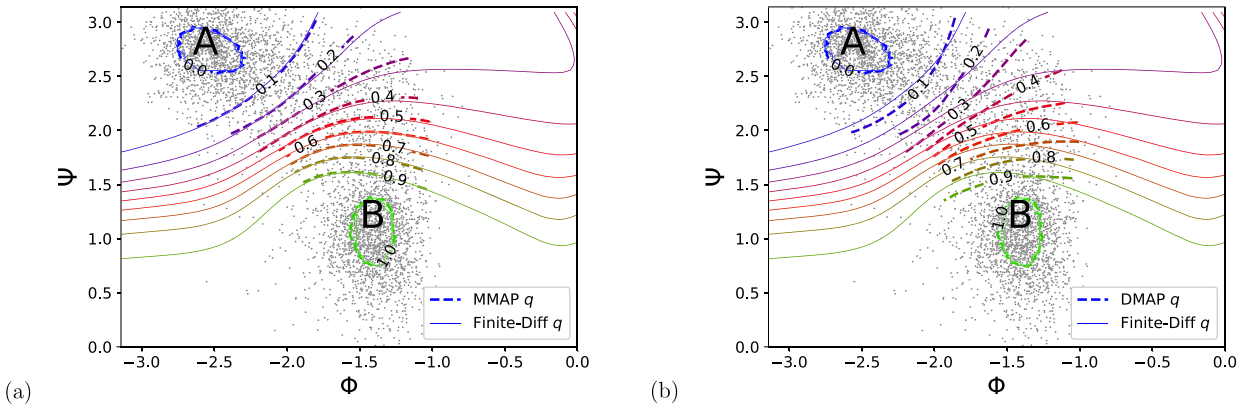

Fig. 5. Level sets for the approximate committor functions obtained from `mmap` (a) and `dmap` (b) on the point cloud (gray dots), with A as the reactant region and B the product region. The dotted lines represent the committor level sets obtained by `mmap` (a) and `dmap` (b), while the solid lines depict the committor level sets obtained by the finite-difference method.

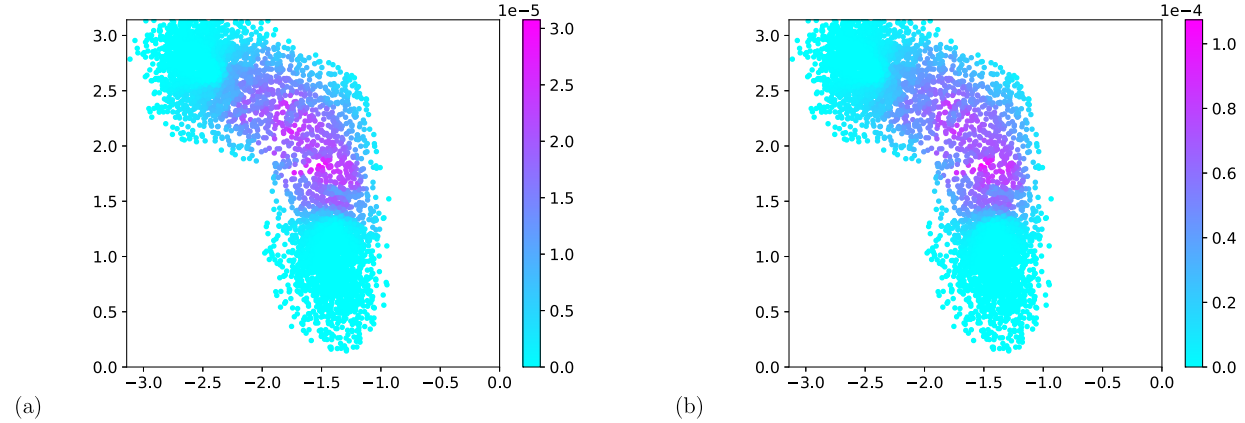

Fig. 6. The intensity of the reactive current computed using the `mmap` (a) and `dmap` (b) committors.

0.014 and 0.036 respectively. We know that the range of ϵ used for `mmap` is shifted with respect to the range of `dmap` due to the fact that the eigenvalues of M^{-1} range from 1.16 to 8.11 and average to 4.06. As a result, the optimal ϵ for `mmap` is approximately larger than that for `dmap` by a factor of 3.33. Moreover, for all ϵ-values in the overlap of ranges the error for `mmap` is smaller than that of `dmap`. The level sets of the computed committor using `mmap` and `dmap` for the values of ϵ minimizing the error are shown, respectively, in Fig. 5(a) and 5(b) with dashed lines. The solid lines are the corresponding level sets of the committor q_{true} computed by finite differences. The level sets of the `mmap` committor closely match those of q_{true}, while the level sets of the `dmap` committor notably deviate from them.

As we have explained in Section 2.2 the committor allows us to compute the reactive current and the transition rate. The calculation of the reactive current and the reaction rate is detailed in Appendix E. The reactive currents computed using the `mmap` and `dmap` committors, respectively, are visualized in Fig. 6 (a) and (b). Notably, the intensity for the respective currents differs by an order of magnitude. The corresponding reaction rates for `mmap` and `dmap` are, respectively, $\nu_{AB} = 0.092 \times 10^{-12}$ s^{-1} and $\nu_{AB} = 0.31 \times 10^{-12}$ s^{-1}. To verify the rate, we ran 10 long trajectories and for each calculated the transition rate as the ratio of the number of transitions from A to B over the elapsed time. The mean rate over the trajectories is $\nu_{AB} = 0.093 \times 10^{-12}$ s^{-1} (standard deviation 0.003×10^{-12} s^{-1}) which is very close to the `mmap` rate and notably differs from the `dmap` rate.

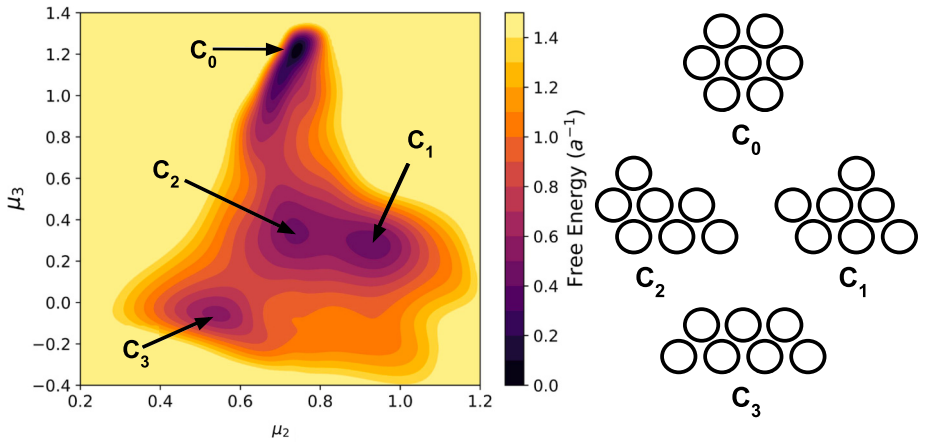

Fig. 7. Free energy surface of LJ7 system with respect to 2nd and 3rd moment of coordination numbers CVs. The four minima C_k, $k = 0, 1, 2, 3$, are marked in the free energy plot and depicted on the right.

4.2. Transitions between the trapezoid and the hexagon in Lennard-Jones 7 in 2D

The cluster of seven 2D particles interacting according to the Lennard-Jones pair potential

$$V_{pair}(r) = 4a\left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6}\right]$$

where $\sigma > 0$ and $a > 0$ are parameters controlling, respectively, range and strength of interparticle interaction, has been another benchmark problem in chemical physics [22,29–31]. If the particles are treated as indistinguishable, the potential energy surface

$$V(x) = \sum_{i<j} V_{pair}(\|x_i - x_j\|), \quad 1 \le i, j \le 7,$$

has four local minima denoted by C_0 (hexagon), C_1 (capped parallelogram 1), C_2 (capped parallelogram 2), and C_3 (trapezoid) – see Fig. 7.

4.2.1. Choosing collective variables

Following [54,55], we chose the 2nd and 3rd central moments of the distribution of coordination numbers as collective variables (CVs). These CVs allow us to separate all four minima in a 2D space. The coordination number of particle i, $1 \le i \le 7$, is a smooth function approximating the number of nearest neighbors of i:

$$c_i(x) = \sum_{j \ne i} \frac{1 - \left(\frac{r_{ij}}{1.5\sigma}\right)^8}{1 - \left(\frac{r_{ij}}{1.5\sigma}\right)^{16}}, \quad \text{where} \quad r_{ij} := \|x_i - x_j\|. \tag{47}$$

Let us elaborate on it. The interparticle distance minimizing $V_{pair}(r)$ is $r^* = 2^{1/6}\sigma$. We would like to treat particles as nearest neighbors if the distance between them is close to r^*. If four particles arranged into a square, the diagonal particles at distance $r^*\sqrt{2} \approx 1.5874\sigma$ should be "not quite" nearest neighbors. Particles at distance $2r^*$ should not count as nearest neighbors. Normalizing the distance to 1.5σ in (47) makes the desired distinction. Indeed, we have:

$$\frac{1 - \left(\frac{r^*}{1.5\sigma}\right)^8}{1 - \left(\frac{r^*}{1.5\sigma}\right)^{16}} \approx 0.91 \sim 1, \quad \frac{1 - \left(\frac{r^*\sqrt{2}}{1.5\sigma}\right)^8}{1 - \left(\frac{r^*\sqrt{2}}{1.5\sigma}\right)^{16}} \approx 0.39, \quad \frac{1 - \left(\frac{2r^*}{1.5\sigma}\right)^8}{1 - \left(\frac{2r^*}{1.5\sigma}\right)^{16}} \approx 0.04 \sim 0.$$

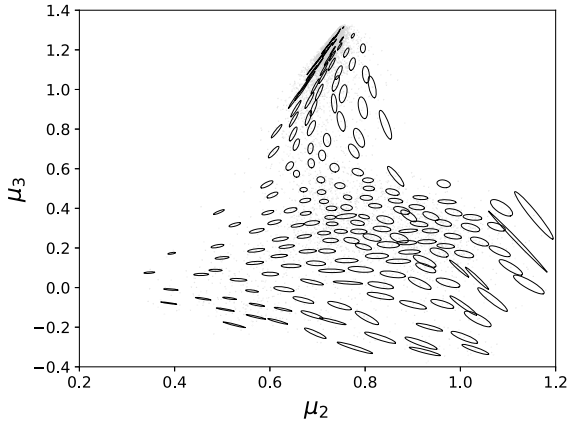

Fig. 8. Ellipses corresponding to the principal components of the estimated diffusion matrices $M(x_j)$ for a subsampled set of the LJ7 dataset depicted with faint gray dots.

The *kth* central moment of $c_i(x)$ is

$$\mu_k(x) := \frac{1}{7}\sum_i (c_i(x) - \bar{c}(x))^k, \quad \text{where} \quad \bar{c}(x) = \frac{1}{7}\sum_j (c_j(x)). \tag{48}$$

The moments μ_k are invariant with respect to permutation of particles, which is important as the particles are indistinguishable. The space of the chosen collective variables (μ_2, μ_3) is $\mathbb{R}^2$.

4.2.2. Obtaining data

We set the temperature for the simulation to $0.2\frac{k_b T}{a}$ and used a Langevin thermostat with relaxation time $0.1\sqrt{\frac{a}{m\sigma^2}}$. To prevent clusters from evaporating, we imposed restraints to keep the atoms from moving further than 2σ from the center of mass from the cluster. Then we simulated the trajectory at timestep $0.005\sqrt{\frac{a}{m\sigma^2}}$ for 10^7 steps using the velocity Verlet algorithm as implemented in the PLUMED software [52]. For use with diffusion maps, we subsampled the trajectory at regular intervals of time to obtain 7500 data points. The diffusion matrices obtained as described in [5] and Appendix A, and are visualized in Fig. 8. The reactant set A and product set B are chosen to be the energy minima C_3 and C_0 respectively. This choice was motivated by the fact that C_3 and C_0 are the most distant pair of metastable states, separated by the highest potential energy barrier [29,30], and connected with a wide transition channel passing through the basins of C_1 and C_2. It is worth noting that such a situation where the region between two stable states of interest is interspersed with other metastable states is quite common in practical situations [56]. Furthermore, transitions between C_0 and C_3 occur very infrequently, and it takes much longer time to accumulate statistics for them in numerical simulation than for the transition in alanine dipeptide considered in Section 4.1.

The scaling parameters ϵ for `mmap` and `dmap` were set, respectively, to

$$\epsilon = \max_i \min_{j\neq i} s(i,j), \quad \text{where} \tag{49}$$

$$s(i,j) := \frac{1}{2}(x_i - x_j)^\top (M^{-1}(x_i) + M^{-1}(x_j))(x_i - x_j) \quad \text{for } \texttt{mmap}, \tag{50}$$

$$s(i,j) := ||x_i - x_j||_2^2 \quad \text{for } \texttt{dmap}. \tag{51}$$

This simple procedure for choosing ϵ worked remarkably well. As with the previous example, we used kNN sparsification for the `mmap` kernel.

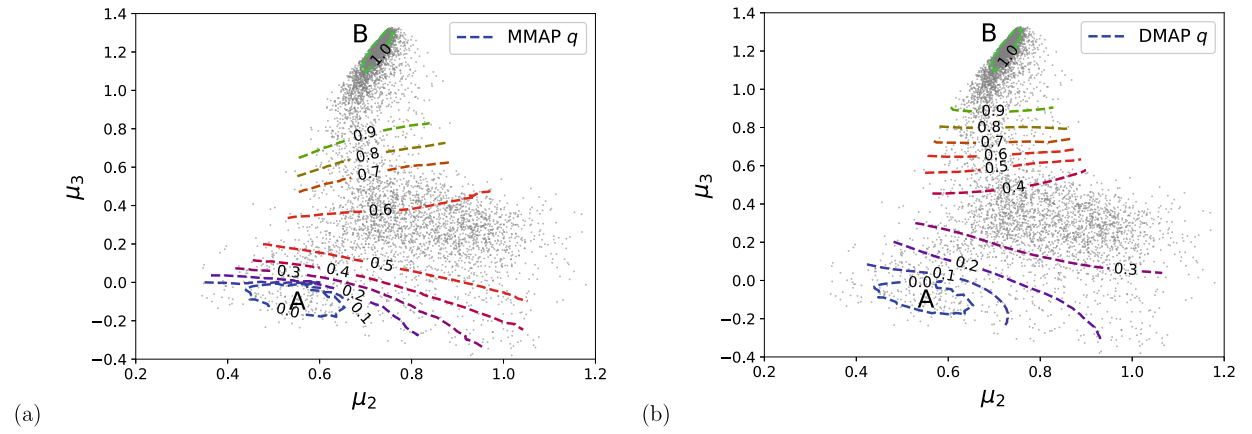

Fig. 9. Level sets for the `mmap` (a) and `dmap` (b) committors are depicted with dashed lines. The set of data points is shown with gray dots.

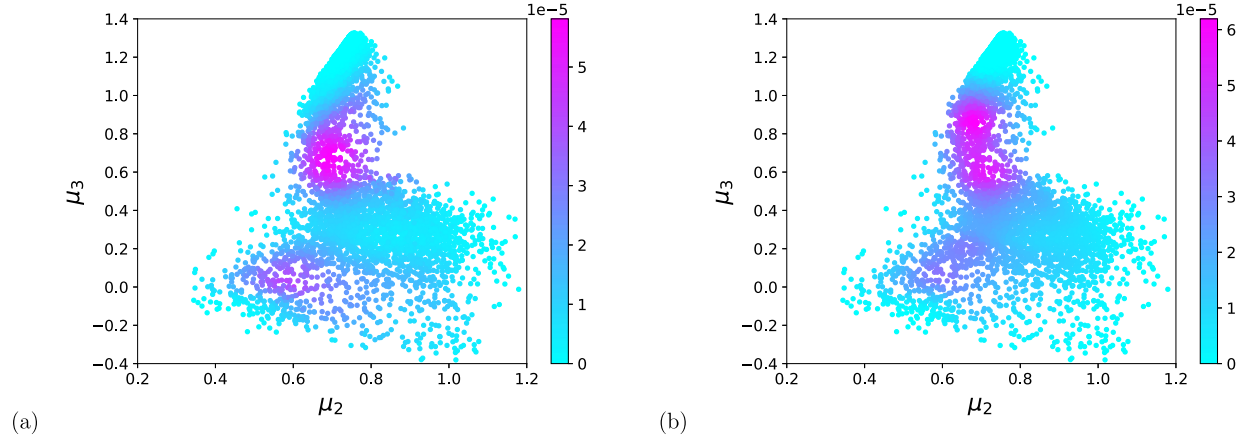

Fig. 10. The intensity of the reactive current obtained with the `mmap` (a) and `dmap` (b) committors.

4.2.3. Results and validation

The level sets of the `mmap` and `dmap` committors are shown in Fig. 9. Notably, these committors significantly differ from each other. In particular, the $q = 0.5$ level sets for the `mmap` and `dmap` committors lie on opposite sides of the dynamical trap surrounding the basins of C_1 and C_2 minima. The reactive currents computed using the `mmap` and `dmap` committors respectively, are displayed in Fig. 10 (a) and (b).

To validate our results and determine which of the `mmap` or `dmap` committors is more accurate, we performed committor analysis [5,32,33], a common statistical validation technique for committors in collective variables. Committor analysis checks a particular committor level set by using the definition of the committor at x as the probability that a stochastic trajectory starting at x first reaches B rather than A. We verified the most important level set $q = 0.5$, the *transition state*. We sampled a set of $N_{pt} = 1000$ points x_j along this level set and launched an ensemble of $N_e = 200$ trajectories from each of them. For each x_j, we counted the number of trajectories N_B that reached first C_0 rather than C_3 and denoted the ratio N_B/N_e by $p_B(x_j)$. We then plotted a histogram with each bin defined by a p_B value and counts determined by the number of selected points in the level set $q = 0.5$ with that p_B value normalized by N_{pt} (Fig. 11). A well-approximated $q = 0.5$ level set should have a unimodal histogram with a sharp peak at $p_B = 0.5$. We see that the distribution for `mmap` peaks at 0.5 as expected, while the `dmap` distribution peaks at 0.75, missing the correct statistical behavior by a large margin. Therefore, we conclude that `mmap` produces a good approximation for the committor while `dmap` gives a qualitatively wrong result.

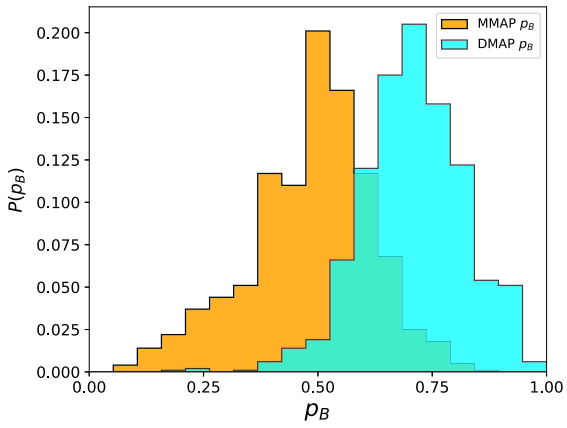

Fig. 11. Committor analysis of the $q = 0.5$ level sets from Fig. 9, with orange corresponding to `mmap` and cyan corresponding to `dmap`.

5. Conclusion

The main conclusion of this work is that the Mahalanobis diffusion map algorithm (`mmap`) is a provably correct, robust and reliable tool for computing the committor in collective variables discretized to a point cloud of data generated by MD simulations. The dynamics in collective variables is governed by a reversible SDE with anisotropic and position-dependent diffusion matrix $M(x)$. The Mahalanobis kernel proposed in [24] accurately captures this anisotropy regardless of whether $M(x)$ is decomposable or not into a product $J(x)J^\top(x)$ where $J(x)$ is the Jacobian matrix for some diffeomorphism.

Specifically, we have calculated the limiting family of differential operators converged to by the α-indexed `mmap` family of matrix operators, where convergence is with respect to the number of data points tending to infinity and scaling parameter ϵ tending to 0. If $\alpha = 1/2$, the limiting operator is the generator for the overdamped Langevin SDE in collective variables. In our derivation, we have discarded the key assumption of [24] that $M(x)$ is associated with a diffeomorphism.

We have chosen two benchmark chemical physics systems as test problems: transitions in alanine dipeptide and rearrangement of an LJ7 cluster of 2D particles. On these examples, we have demonstrated that `mmap` is easy to implement and gives good results for any reasonable choice of the scaling parameter epsilon. We have validated our results by comparing the committors computed with `mmap` to the one obtained using a traditional finite difference method or by conducting committor analysis. We have contrasted the committor by `mmap` with the one by the diffusion map with isotropic Gaussian kernel and shown that the latter can lead to a wrong placement of the transition state and highly inaccurate estimate for the reaction rate.

In the current setting, `mmap` has a significant limitation: it requires the input data to be sampled from the invariant probability density. This prevents us from using enhanced sampling techniques such as temperature acceleration [57] and metadynamics [58] which are standard techniques applied to promote transitions between metastable states in MD simulations. We plan to address this problem in our future work.

Data availability

The code used for diffusion maps will be shared at "https://github.com/aevans1/targetmeasure-mmap".

Acknowledgments

We thank Prof. Y. Kevrekidis for encouraging us to look into Mahalanobis diffusion maps and for providing valuable feedback on this manuscript. We thank Prof. T. Berry for a valuable discussion on whether every symmetric positive definite matrix function is associated with a variable change. We are grateful

to the anonymous reviewers for their thorough reviews and helpful suggestions. This work was partially supported by NSF CAREER grant DMS-1554907 (MC), AFOSR MURI grant FA9550-20-1-0397 (MC), and NSF CAREER grant CHE-2044165 (PT). Computing resources Deepthought2, MARCC and XSEDE (project TG-CHE180053) were used to generate molecular simulation data.

Appendix A. The free energy and the diffusion tensor

The free energy $F(x)$ and the diffusion matrix $M(x)$ in SDE (1) are defined, respectively, as follows:

$$F(x) = -\beta^{-1} \ln \left(\int_{\mathbb{R}^m} Z_V^{-1} e^{-\beta V(y)} \prod_{l=1}^{d} \delta(\theta_l(y) - x_l) dy \right), \tag{A.1}$$

$$M(x) = e^{-\beta F(x)} \int_{\mathbb{R}^m} J(y) J^\top(y) Z_V^{-1} e^{-\beta V(y)} \prod_{l=1}^{d} \delta(\theta_l(y) - x_l) dy. \tag{A.2}$$

In (A.2), $J(x)$ is the Jacobian matrix whose entries are

$$J_{ij}(y) = \frac{\partial \theta_i(y)}{\partial y_j} \qquad 1 \le i \le d, \quad 1 \le j \le m.$$

To apply the `mmap` algorithm, we do not need to know the free energy. However, we do need evaluate the diffusion matrix $M(x)$ at the data points. A method for estimating the diffusion matrix $M(x)$ of (A.2) was described in [5]. Here we outline it for the reader's convenience.

First, we approximate the distribution $\prod_{l=1}^{d} \delta(\theta_l(y) - x_l)$ with a Gaussian. We fix $x \in \mathbb{R}^d$, choose a large spring constant $\kappa > 0$, and consider a constrained system with the "extended potential" given by

$$U(y; \kappa, x) = V(y) + \frac{\kappa}{2} ||\theta(y) - x||^2, \tag{A.3}$$

evolving according to the overdamped Langevin dynamics

$$\begin{aligned} dy_t &= -\nabla U(y_t; \kappa, x) dt + \sqrt{2\beta^{-1}} dW_t \\ &= \left[-\nabla V(y) - \kappa J(y)^\top (\theta(y) - x) \right] dt + \sqrt{2\beta^{-1}} dW_t. \end{aligned} \tag{A.4}$$

The restrained dynamics (A.4) has stationary distribution

$$\rho(y; \kappa, x) := Z(\kappa, x)^{-1} e^{-\beta U(y;\kappa,x)} \quad \text{where} \quad Z(\kappa, x) := \int_{\mathbb{R}^d} e^{-\beta U(y;\kappa,x)} dy,$$

with limiting distribution

$$\lim_{\kappa \to \infty} \int_{\mathbb{R}^d} f(y) \rho(y; \kappa, x) dy = \int_{\mathbb{R}^d} f(y) Z_V^{-1} e^{-\beta U(y)} \prod_{l=1}^{d} \delta(\theta_l(y) - x_l) dy$$

Next, we generate trajectory data $\{y_{t_i}\}_{i=0}^n$ for the restrained dynamics (A.4). These data enable us to estimate the conditional expectation for an arbitrary function f as follows:

$$\lim_{\kappa\to\infty}\lim_{n\to\infty}\frac{1}{n}\sum_{i=1}^{n}f(y_{t_i}) = \lim_{\kappa\to\infty}\lim_{n\to\infty}\frac{1}{n\Delta t}\int_0^{n\Delta t} f(y_t)\rho(y_t;\kappa,x)dt$$
$$= e^{\beta F(x)}\int_{\mathbb{R}^n} f(y)Z_V^{-1}e^{-\beta V(y)}\prod_{l=1}^{d}\delta(\theta_l(y)-x_l)dy$$
$$= \mathbb{E}[f|\theta(y)=x]. \tag{A.5}$$

In particular, the diffusion tensor $M(x)$ for the collective variables $x=\theta(y)$ is estimated according the formula:

$$M_{ij}(x) \approx \frac{1}{n}\sum_{i=1}^{n}\sum_{l=1}^{m}\frac{\partial\theta_i(y_{t_i})}{\partial y_l}\frac{\partial\theta_j(y_{t_i})}{\partial y_l}. \tag{A.6}$$

The mean force, i.e., the gradient of the free energy $\nabla F(x)$, also can be estimated in a similar manner:

$$\nabla F(x) = \frac{\kappa}{n}\sum_{i=1}^{n}\left(x-\theta(y_{t_i})\right). \tag{A.7}$$

We evaluate $M(x)$ by (A.6) in both applications presented in this work. This procedure is an outgrowth of well-established uses for constrained dynamics within the molecular dynamics community, particularly in fundamental works for computing free energy differences [59,60] and position-dependent friction [61].

For a general diffusion matrix not necessarily of form (A.2) or for when the Jacobian $J(y)$ is not available, one can utilize local covariances as described in [24,38,41,44,47]. These approaches utilize that for (1),

$$M(x) = \lim_{\Delta t\to 0}\frac{\mathbb{E}\left[(x_{t+\Delta t}-x_t)(x_{t+\Delta t}-x_t)^\top|x_t=x\right]}{2\beta^{-1}\Delta t},$$

and derive estimators of $M(x)$ by approximating the right-hand side through short simulation bursts initiated at x or from small neighborhoods of the trajectory near x.

Appendix B. Proof of Theorem 3.1

We will need two auxiliary lemmas. The first lemma gives a necessary condition for a matrix-function to be Jacobian.

Lemma B.1. *Let $z=f(x)$ be a twice continuously differentiable coordinate change $f:\mathbb{R}^d\to\mathbb{R}^d$ with Jacobian matrix*

$$J(x) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_d} \\ \vdots & & \vdots \\ \frac{\partial f_d}{\partial x_1} & \cdots & \frac{\partial f_d}{\partial x_d} \end{bmatrix}. \tag{B.1}$$

Then, the entries of J satisfy

$$\frac{\partial J_{ij}}{\partial x_k} = \frac{\partial J_{ik}}{\partial x_j} \quad \forall 1\le i,j,k\le d,\ j\neq k. \tag{B.2}$$

Proof. Indeed, the left and right-hand side of (B.2) are the mixed partials that are equal as they are continuous:

$$\frac{\partial J_{ij}}{\partial x_k} = \frac{\partial^2 f_i}{\partial x_k \partial x_j} = \frac{\partial^2 f_i}{\partial x_j \partial x_k} = \frac{\partial J_{ik}}{\partial x_j}. \quad \square$$

The second lemma shows that any decomposition $M = AA^\top$ of a symmetric positive definite matrix relates to $M^{1/2}$ via an orthogonal transformation.

Lemma B.2. *Let M be a symmetric positive definite matrix, and A be any matrix such that $M = AA^\top$. Then, there exists an orthogonal transformation O such that $A = M^{1/2}O$.*

Proof. We have:

$$M = M^{1/2}M^{1/2} = AA^\top.$$

Multiplying this identity by $M^{-1/2}$ on the right and on the left we get:

$$I = M^{-1/2}AA^\top M^{-1/2} = M^{-1/2}A\left(M^{-1/2}A\right)^\top.$$

Hence $O := M^{-1/2}A$ is orthogonal. Therefore, $A = M^{1/2}O$ as desired. $\square$

Proof of Theorem 3.1. First we prove that (38) *is necessary for the existence of decomposition* (36). We observe that $M^{1/2}(x,y) = m(x,y)I_{2\times 2}$ is a Jacobian of a vector-function $f : \Omega \to \mathbb{R}^2$ if and only if $m(x,y)$ is constant. Indeed, condition (B.2) applied to $M^{1/2}(x,y) = m(x,y)I_{2\times 2}$ reduces to $m_y = 0$ and $m_x = 0$. Note that any constant function is harmonic.

Suppose that $m(x,y)$ is not constant. In this case, by Lemma B.2, if M admits decomposition (36) then $J(x,y)$ must be of the form $M^{1/2}(x,y)O(x,y)$ for some orthogonal matrix $O(x,y)$. There are two families of orthogonal 2×2 matrix functions:

$$O(x,y) = \begin{bmatrix} \cos\phi(x,y) & \sin\phi(x,y) \\ -\sin\phi(x,y) & \cos\phi(x,y) \end{bmatrix} \quad \text{and} \quad O(x,y) = \begin{bmatrix} \cos\phi(x,y) & \sin\phi(x,y) \\ \sin\phi(x,y) & -\cos\phi(x,y) \end{bmatrix}. \tag{B.3}$$

Hence, $M^{1/2}(x,y)O(x,y)$ is of the form

$$m(x,y)\begin{bmatrix} \cos\phi(x,y) & \sin\phi(x,y) \\ -\sin\phi(x,y) & \cos\phi(x,y) \end{bmatrix} \quad \text{or} \quad m(x,y)\begin{bmatrix} \cos\phi(x,y) & \sin\phi(x,y) \\ \sin\phi(x,y) & -\cos\phi(x,y) \end{bmatrix}.$$

Condition (B.2) applied to $M^{1/2}(x,y)O(x,y)$ requires the following equalities to hold:

$$\frac{\partial}{\partial y}(m\cos\phi) = \frac{\partial}{\partial x}(m\sin\phi),$$
$$\frac{\partial}{\partial y}(m\sin\phi) = -\frac{\partial}{\partial x}(m\cos\phi).$$

Performing differentiation, we get:

$$m_y\cos\phi - m\phi_y\sin\phi = m_x\sin\phi + m\phi_x\cos\phi$$
$$m_y\sin\phi + m\phi_y\cos\phi = -(m_x\cos\phi - m\phi_x\sin\phi)$$

Regrouping the terms, we obtain:

$$m_y \cos\phi - m_x \sin\phi = m\left[\phi_x \cos\phi + \phi_y \sin\phi\right] \tag{B.4}$$

$$-m_y \sin\phi - m_x \cos\phi = m\left(\phi_y \cos\phi - \phi_x \sin\phi\right). \tag{B.5}$$

The last set of identities can be rewritten in a matrix form:

$$\begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} m_y \\ -m_x \end{bmatrix} = m \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \begin{bmatrix} \phi_x \\ \phi_y \end{bmatrix} \tag{B.6}$$

The matrix in (B.6) is orthogonal. Hence, multiplying (B.6) by its transpose we get:

$$\frac{m_y}{m} \equiv [\log m]_y = \phi_x, \quad -\frac{m_x}{m} \equiv -[\log m]_x = \phi_y. \tag{B.7}$$

The fact that the mixed partials of ϕ must be equal implies that

$$[\log m]_{xx} + [\log m]_{yy} = 0. \tag{B.8}$$

This completes the proof that (38) is necessary for the existence of decomposition (36).

Next, we prove that (38) *is sufficient for the existence of decomposition* (36) *if* Ω *is simply connected.* This immediately follows from the theorem of calculus saying that if the components of a two-dimensional continuously differentiable vector field $[p, q]^\top$ satisfy $p_y = q_x$ in a simply connected domain Ω then this vector field is conservative. Indeed, we define the vector field $[p, q]^\top$ by $p = [\log m]_y$ and $q = -[\log m]_x$. This vector field satisfies the condition $p_y = q_x$ as $\log m$ is harmonic. We choose a point $(x_0, y_0) \in \Omega$, fix it, and for any other point $(x, y) \in \Omega$ choose a path $\gamma \subset \Omega$ from (x_0, y_0) to (x, y) and integrate the vector field (p, q) along it (see the integral in the right-hand side of equation (B.10) below). We claim that the value of this integral is independent of the path γ from (x_0, y_0) to (x, y). Indeed, if γ' is some other path connecting these points, then we define a closed contour by reversing the path γ' and apply Green's theorem

$$\oint_C p dx + q dy = \iint_U (p_y - q_x) dx dy, \tag{B.9}$$

where U is the region bounded by the simple closed contour C. Since $p_y - q_x$ is identically zero, the contour integral also must be zero. If the contour formed by the paths γ and the reverse of γ' is self-intersecting, we apply Green's theorem to all simple closed contours formed by these paths. Therefore, we can define a function $\phi(x, y)$ by

$$\phi(x, y) = \int_\gamma [\log m]_y dx - [\log m]_x dy \tag{B.10}$$

where $\gamma \subset \Omega$ is any path from (x_0, y_0) to (x, y) set $J(x, y) = m(x, y)O(x, y)$ where $O(x, y)$ is any orthogonal matrix function of the form (B.3) with ϕ defined by (B.10).

Finally, we show that *if* Ω *is not simply connected, the condition* (38) *is not sufficient for the existence of decomposition* (36). We adapt the famous counterexample of a non-conservative vector field. Consider the vector field

$$\begin{bmatrix} p \\ q \end{bmatrix} = \frac{a}{x^2 + y^2} \begin{bmatrix} -y \\ x \end{bmatrix}, \tag{B.11}$$

where $a \neq 0$ is a constant, with the property that $p_y = q_x$. It is smooth in $\mathbb{R}^2 \backslash \{(0, 0)\}$. Let

$$m(x,y) = \left(x^2+y^2\right)^{-a/2}. \tag{B.12}$$

It is easy to check that $[\log m]_y = p$ and $-[\log m]_x = q$ and hence $\log m$ is harmonic everywhere except for the origin. Let us choose $\phi(x,y)$ satisfying $\phi_x = p$ and $\phi_y = q$ to be[2]

$$\phi(x,y) = \begin{cases} a\arctan\left(y/x\right), & x>0 \\ \frac{\pi a}{2}, & x=0 \\ \pi a + a\arctan\left(y/x\right), & x<0 \end{cases} \tag{B.13}$$

The function ϕ is smooth everywhere except for the origin and the negative y-axis. It has a jump discontinuity of size $2\pi a$ along the negative y-axis. If $a \notin \mathbb{Z}$, the functions $\sin\phi$ and $\cos\phi$ will be discontinuous along the negative y-axis. Hence decomposition (36) does not exist. □

Remark B.1. However, if $a \in \mathbb{Z}$ in (B.12), the orthogonal matrices (B.3) with ϕ given by (B.13) are smooth everywhere except for the origin. In particular, if $a = 1$, we have:

$$\frac{1}{\sqrt{x^2+y^2}}\begin{bmatrix} x & y \\ -y & x \end{bmatrix} \quad \text{and} \quad \frac{1}{\sqrt{x^2+y^2}}\begin{bmatrix} x & y \\ y & -x \end{bmatrix} \tag{B.14}$$

Appendix C. Proof of Theorem 3.3

We fix $r \geq 0$, $x \in \mathbb{R}^d$, and a symmetric positive definite matrix $A \in \mathbb{R}^{d\times d}$. Then $\mathcal{B}_r(x;A)$ denotes the ellipsoid

$$\mathcal{B}_r(x;A) := \{y \in \mathbb{R}^d \mid (y-x)^\top A(y-x) \leq r^2\}.$$

The proof of Theorem 3.3 includes two technical lemmas.

Lemma C.1. *Let $x \in \mathbb{R}^d$ be fixed, $\phi(x+a)$ be a function that grows not faster than a polynomial as $\|a\| \to \infty$, and A be a positive definite matrix. Then, for all small enough $\epsilon > 0$ and any $\mu \in (0, 1/2)$, we have:*

$$I := \left| \int\limits_{\mathbb{R}^d \setminus \mathcal{B}_{\epsilon^\mu}(x;A)} e^{-\frac{1}{2\epsilon}(y-x)^\top A(y-x)}\phi(y)dy \right| \leq \epsilon^{d/2} p(\epsilon^{2\mu-1}) e^{-\frac{\epsilon^{2\mu-1}}{2}}, \tag{C.1}$$

where p is a polynomial.

Proof. We implement two variable changes. First we introduce $z := \epsilon^{-1/2}A^{1/2}(y-x)$ and then switch to spherical coordinates in $\mathbb{R}^d$. We calculate:

$$\begin{aligned} I &= \left| \int\limits_{\mathbb{R}^d \setminus \mathcal{B}_{\epsilon^\mu}(x;A)} e^{-\frac{1}{2\epsilon}(y-x)^\top A(y-x)}\phi(y)dy \right| \\ &= \left| \int\limits_{\mathbb{R}^d \setminus \mathcal{B}_{\epsilon^{2\mu-1}}(0;I)} e^{-\frac{1}{2}\|z\|^2}\phi(x+\sqrt{\epsilon}A^{-1/2}z)\epsilon^{d/2}|A|^{-1/2}dz \right| \end{aligned}$$

[2] Acknowledgment of lecture notes by E. L. Lady: http://www.math.hawaii.edu/~lee/calculus/potential.pdf.

Note that $2\mu-1<0$, hence $\epsilon^{2\mu-1}\to\infty$ as $\epsilon\to 0$. Further, since $\phi(x+a)$ grows not faster than a polynomial as $\|a\|\to\infty$, there are constants C_1 and C_2 and a positive integer k such that

$$\left|\left(\phi(x)+\sqrt{\epsilon}A^{-1/2}z\right)\right|\leq C_1+\sqrt{\epsilon}C_2\lambda_{\max}(A^{-1/2})\|z\|^k.$$

Therefore, aiming at switching to spherical coordinates in $\mathbb{R}^d$, we write:

$$\left|\|z\|^{d-1}\left(\phi(x)+\sqrt{\epsilon}A^{-1/2}z\right)\right|\leq C_1\|z\|^{d-1}+\sqrt{\epsilon}C_2\lambda_{\max}(A^{-1/2})\|z\|^{k+d-1}\leq C\|z\|^m,$$

where C is some constant, and m is the smallest odd integer greater or equal to $d+k-1$. Finally, switching to spherical coordinates and denoting the surface of the $(d-1)$-dimensional unit sphere by $|S_{d-1}|$, we derive the desired estimate:

$$\begin{aligned}I\leq&\frac{C\epsilon^{d/2}|S_{d-1}|}{|A|^{1/2}}\int\limits_{\epsilon^{\mu-1/2}}^{\infty}e^{-\frac{r^2}{2}}r^m dr\\=&\frac{C\epsilon^{d/2}|S_{d-1}|}{|A|^{1/2}}\int\limits_{\frac{\epsilon^{2\mu-1}}{2}}^{\infty}e^{-t}t^{\frac{m-1}{2}}dt\\=&\epsilon^{d/2}p(\epsilon^{2\mu-1})e^{-\frac{\epsilon^{2\mu-1}}{2}},\end{aligned}$$

where the polynomial p is obtained from integrating by parts $(m-1)/2$ times and multiplying the result by $C|A|^{-1/2}$. □

Lemma C.2. *Let $\mathcal{G}_\epsilon$ be an integral operator defined by*

$$\mathcal{G}_\epsilon f(x)=\int\limits_{\mathbb{R}^d}e^{-\frac{1}{4\epsilon}(x-y)^\top[M^{-1}(x)+M^{-1}(y)](x-y)}f(y)dy,\tag{C.2}$$

where the matrix function M and the scalar function f satisfy Assumption 2. Let $\tilde{M}\equiv M^{-1}$ and

$$\tilde{M}(y)=\tilde{M}(x)+\nabla\tilde{M}(x)(y-x)+r_2(x;y-x)+r_3(x;y-x)+O(\|y-x\|^4),\tag{C.3}$$

where $\nabla\tilde{M}(x)(y-x)$ is a matrix with entries

$$\left(\nabla\tilde{M}(x)(y-x)\right)_{ij}=\nabla\tilde{M}_{ij}^\top(y-x),$$

and $r_2(x;z)$ and $r_3(x;z)$ are the matrices whose entries are the second and third-order terms in Taylor expansions of $\tilde{M}_{ij}$. Then

$$\begin{aligned}\mathcal{G}_\epsilon f(x)=\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}\Big(f(x)+\epsilon\Big[&-\nabla f(x)^\top\omega_1(x)-f(x)\omega_2(x)\\&+\frac{1}{2}\mathsf{tr}(\tilde{M}(x)^{-1}H(x))\Big]+O(\epsilon^2)\Big),\end{aligned}\tag{C.4}$$

where $H(x):=\nabla\nabla f(x)$ is the Hessian matrix for f evaluated at x, and

$$\omega_{1,i}(x) := \frac{|\tilde{M}|^{1/2}}{(2\pi\epsilon)^{d/2}} \frac{1}{4\epsilon^2} \int\limits_{\mathbb{R}^d} e^{-\frac{z^\top \tilde{M} z}{2\epsilon}} z_i \left[z^\top [\nabla \tilde{M}(x) z] z \right] dz, \quad 1 \le i \le d, \tag{C.5}$$

$$\omega_2(x) := \frac{|\tilde{M}|^{1/2}}{(2\pi\epsilon)^{d/2}} \int\limits_{\mathbb{R}^d} e^{-\frac{z^\top \tilde{M} z}{2\epsilon}} \left[\frac{z^\top r_2(x;z) z}{4\epsilon^2} - \frac{\left(z^\top [\nabla \tilde{M}(x) z] z \right)^2}{32\epsilon^3} \right] dz. \tag{C.6}$$

Proof. The proof utilizes the Taylor expansion $f(x+z)$ around $f(x)$:

$$f(x+z) = f(x) + \nabla f(x)^\top z + \frac{1}{2} z^\top \nabla\nabla f(x) z + p_3(z) + p_4(z), \tag{C.7}$$

where $p_3(z)$ is a homogeneous third degree polynomial in z, and $p_4(z)$ is $O(z^4)$. To establish (C.4), we will need to integrate the products of each of these terms with the Mahalanobis kernel

$$k_\epsilon(x,y) = e^{-\frac{1}{4\epsilon}(x-y)^\top [\tilde{M}(x) + \tilde{M}(y)](x-y)}. \tag{C.8}$$

First we eliminate the dependence of the matrix $\tilde{M}$ on the integration variable y in the exponent by using Taylor expansions. Let $z := y - x$ and ϱ be the residual in the expansion (C.3):

$$\varrho(x, x+z) := \tilde{M}(x+z) - \left\{ \tilde{M}(x) + \nabla \tilde{M}(x) z + r_2(x;z) + r_3(x;z) \right\} = O(\|z\|^4).$$

Then

$$e^{-\frac{(y-x)^\top [\tilde{M}(x) + \tilde{M}(y)](y-x)}{4\epsilon}} = e^{-\frac{z^\top \tilde{M}(x) z}{2\epsilon}} e^{-\frac{z^\top [\nabla \tilde{M}(x) z + r_2(x;z) + r_3(x;z) + \varrho(x,y)] z}{4\epsilon}}. \tag{C.9}$$

The Taylor series for $\exp(-t)$ converges on $\mathbb{R}$. Hence, expanding the second exponent in (C.9) we get:

$$\begin{aligned}
& e^{-\frac{z^\top [\nabla \tilde{M}(x) z + r_2(z) + r_3(z) + \varrho(x,y)] z}{4\epsilon}} \\
=& 1 - \frac{1}{4\epsilon} \left(z^\top [\nabla \tilde{M}(x) z + r_2(x;z) + r_3(x;z) + \varrho(x,y)] z \right) \\
+& \frac{1}{32\epsilon^2} \left(z^\top [\nabla \tilde{M}(x) z + r_2(x;z) + r_3(x;z) + \varrho(x,y)] z \right)^2 - \dots \\
+& \frac{(-1)^k}{k!(4\epsilon)^k} \left(z^\top [\nabla \tilde{M}(x) z + r_2(x;z) + r_3(x;z) + \varrho(x,y)] z \right)^k + \dots .
\end{aligned}$$

Second, we will split the integral of each term in (C.7) multiplied by $k_\epsilon(x,y)$ into the sum

$$\int\limits_{\mathbb{R}^d} = \int\limits_{\mathcal{B}_{x,\epsilon^\mu}} + \int\limits_{\mathbb{R}^d \setminus \mathcal{B}_{x,\epsilon^\mu}},$$

where $\mathcal{B}_{x,\epsilon^\mu}$ denotes the ellipse $\mathcal{B}_{\epsilon^\mu}(x; \tilde{M})$ and $\mu \in (0, 1/2)$ is fixed. Where appropriate, we will apply Lemma C.1 to the integral over $\mathbb{R}^d \setminus \mathcal{B}_{x,\epsilon^\mu}$. We will need the following ingredients.

Integral 1:

$$\begin{aligned}
& \int\limits_{\mathcal{B}_{x,\epsilon^\mu}} k_\epsilon(x,y) dy \\
&= \int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top \tilde{M}(x) z}{2\epsilon}} \left[1 - \frac{1}{4\epsilon} \left(z^\top [\nabla \tilde{M}(x) z + r_2(x;z) + r_3(x;z) + O(\|z\|^4)] z \right) \right.
\end{aligned} \tag{C.10}$$

$$+\frac{1}{32\epsilon^2}\left(z^\top[\nabla\tilde{M}(x)z+r_2(x;z)+r_3(x;z)+\rho(x,y)]z\right)^2-\ldots$$
$$+\frac{(-1)^k}{k!(4\epsilon)^k}\left(z^\top[\nabla\tilde{M}(x)z+r_2(x;z)+r_3(x;z)+\rho(x,y)]z\right)^k+\ldots\Bigg]dz.$$

We will tackle this integral term-by-term. For brevity, we will omit the argument (x) of $\tilde{M}$. Thus,

$$\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}dz=\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}(1+\delta_1(\epsilon)),$$

where $\delta_1(\epsilon)$ decays exponentially fast as $\epsilon\to 0$. Further,

$$\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}\frac{\left(z^\top[\nabla\tilde{M}z]z\right)^{2k+1}}{(4\epsilon)^{2k+1}}dz=0,\quad k=0,1,2,\ldots;$$

$$\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}\frac{\left(z^\top[\nabla\tilde{M}z]z\right)^{2k}}{(4\epsilon)^{2k}}dz=\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}O\left(\epsilon^k\right),\quad k=1,2,\ldots;$$

$$\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}\frac{\left(z^\top r_2(x;z)z\right)^{k}}{(4\epsilon)^{k}}dz=\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}O\left(\epsilon^{2k-1}\right),\quad k=1,2,\ldots;$$

$$\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}\frac{\left(z^\top[\nabla\tilde{M}z]z\right)^{2k}z^\top r_2(x;z)z}{(4\epsilon)^{2k+1}}dz=\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}O\left(\epsilon^{k+1}\right),\quad k=1,2,\ldots.$$

The rest of the integrals originating from (C.10) will be either zero or $O(\epsilon^{d/2}\epsilon^k)$ for some $k>2$. Putting the integrals together and organizing them according to the order in powers of ϵ, we get:

$$\int\limits_{\mathcal{B}_{x,\epsilon^\mu}} k_\epsilon(x,y)dy=\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}\left[1-\epsilon\omega_2(x)+O(\epsilon^2)\right],\tag{C.11}$$

where

$$\omega_2(x):=\frac{|\tilde{M}|^{1/2}}{(2\pi\epsilon)^{d/2}}\int\limits_{\mathbb{R}^d} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}\left[\frac{z^\top r_2(x;z)z}{4\epsilon^2}-\frac{\left(z^\top[\nabla\tilde{M}z]z\right)^2}{32\epsilon^3}\right]dz.\tag{C.12}$$

Note that the value ω_2 is of the order of 1. We have applied Lemma C.1 to replace the integral over $\mathcal{B}_{x,\epsilon^\mu}$ with the one over $\mathbb{R}^d$.

Integral 2:

$$\begin{aligned}&\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} k_\epsilon(x,x+z)z_i dz\\ =&\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}}z_i\left[1-\frac{1}{4\epsilon}\left(z^\top[\nabla\tilde{M}z+r_2(z)+r_3(z)+O(\|z\|^4)]z\right)+\ldots\right]dz\\ =&\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}\left[-\epsilon\omega_{1,i}(x)+O(\epsilon^2)\right],\end{aligned}\tag{C.13}$$

where, with the aid of Lemma C.1,

$$\omega_{1,i}(x) := \frac{|\tilde{M}|^{1/2}}{(2\pi\epsilon)^{d/2}} \frac{1}{4\epsilon^2} \int\limits_{\mathbb{R}^d} e^{-\frac{z^\top \tilde{M} z}{2\epsilon}} z_i \left[z^\top [\nabla \tilde{M} z] z\right] dz, \quad 1 \le i \le d. \tag{C.14}$$

Integral 3:

$$\begin{aligned}
&\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} k_\epsilon(x, x+z) z^\top H z dz \\
=&\int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top \tilde{M} z}{2\epsilon}} z^\top H z \left[1 - \frac{1}{4\epsilon}\left(z^\top [\nabla \tilde{M}(x) z + r_2(z) + r_3(z) + O(\|z\|^4)] z\right)\right] dz \\
=&\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}} \left[\epsilon \mathsf{tr}(\tilde{M}^{-1} H) + O(\epsilon^2)\right].
\end{aligned} \tag{C.15}$$

Using Integrals 1, 2, and 3, we calculate:

$$\mathcal{G}_\epsilon f(x) = \int\limits_{\mathbb{R}^d} e^{-\frac{z^\top [\tilde{M}(x) + \tilde{M}(x+z)] z}{4\epsilon}} f(x+z) dz = \int\limits_{\mathcal{B}_{0,\epsilon^\mu}} [\ldots] dz + \int\limits_{\mathbb{R}^d \setminus \mathcal{B}_{0,\epsilon^\mu}} [\ldots] dz.$$

The second integral in the right-hand side decays exponentially as $\epsilon \to 0$ by Lemma C.1. Therefore, we will incorporate its value into $O(\epsilon^2)$ term below. We continue, omitting the argument x for brevity and recalling that $\tilde{M}^{-1} \equiv M$:

$$\begin{aligned}
\mathcal{G}_\epsilon f =& \int\limits_{\mathcal{B}_{0,\epsilon^\mu}} e^{-\frac{z^\top [\tilde{M}(x) + \tilde{M}(x+z)] z}{2\epsilon}} \left[f + \nabla f^\top z + \frac{1}{2} z^\top \nabla\nabla f z + \ldots\right] dz + \int\limits_{\mathbb{R}^d \setminus \mathcal{B}_{0,\epsilon^\mu}} [\ldots] dz \\
=& (2\pi\epsilon)^{d/2} |M|^{1/2} \left(f + \epsilon \left[-\nabla f^\top \omega_1 - f\omega_2 + \frac{1}{2}\mathsf{tr}(M \nabla\nabla f)\right] + O(\epsilon^2)\right). \quad \square
\end{aligned}$$

Now we prove Theorem 3.3.

Proof. To carry out the proof of Theorem 3.3, we need to calculate the limit

$$\lim_{\epsilon \to 0} \mathcal{L}_{\epsilon,\alpha} f(x) \equiv \lim_{\epsilon \to 0} \frac{\mathcal{P}_{\epsilon,\alpha} f(x) - f(x)}{\epsilon} \tag{C.16}$$

for any fixed $x \in \mathcal{M}$. Central to the calculation of $\mathcal{P}_{\epsilon,\alpha} f(x)$ is the calculation of integrals over $\mathcal{M}$ which is done by splitting each integral into the sum

$$\int\limits_{\mathcal{M}} = \int\limits_{\mathcal{B}_{x,\epsilon^\mu}} + \int\limits_{\mathcal{M} \setminus \mathcal{B}_{x,\epsilon^\mu}},$$

where $\mu \in (0, 1/2)$ meaning that $\epsilon^\mu \to 0$ as $\epsilon \to 0$. Since the manifold $\mathcal{M}$ is either $\mathbb{R}^d$ or $\mathbb{T}^k \times \mathbb{R}^{d-k}$ for some $1 \le 1 \le d$ with the Euclidean metric within any open ball of radius R_{Euc} (41), for the purpose of integration over it, $\mathcal{M}$ can be treated either as $\mathbb{R}^d$ or as a hyperstrip or a hyperbox in $\mathbb{R}^d$ (See Assumption 1). Therefore, if ϵ is small enough so that the whole ellipse $\mathcal{B}_{x,\epsilon^\mu}$ lies within a ball of radius R_{Euc}, for each integral with an integrand satisfying the assumptions of Lemma C.1 we have:

$$\int_{\mathcal{M}} = \int_{\mathcal{B}_{x,\epsilon^\mu}} + \int_{\mathcal{M}\setminus\mathcal{B}_{x,\epsilon^\mu}} \quad \text{and} \quad \left| \int_{\mathcal{M}\setminus\mathcal{B}_{x,\epsilon^\mu}} \right| \leq \left| \int_{\mathbb{R}^d\setminus\mathcal{B}_{x,\epsilon^\mu}} \right|.$$

According to Lemma C.1, this last integral over $\mathbb{R}^d\setminus\mathcal{B}_{x,\epsilon^\mu}$ decays exponentially as $\epsilon \to 0$. Therefore, the integrals over $\mathcal{M}\setminus\mathcal{B}_{x,\epsilon^\mu}$ do not affect the limit (C.16), i.e., (C.16) is completely determined by the integrals over the ellipse $\mathcal{B}_{x,\epsilon^\mu}$ which are the same whether $\mathcal{M}$ is $\mathbb{R}^d$ or $\mathbb{T}^k \times \mathbb{R}^{d-k}$ provided that it satisfies Assumption 1. Hence, Lemma C.2 remains valid for the manifold $\mathcal{M}$.

We will omit the argument x in the calculations within this proof to shorten expressions. Lemma C.2 implies that

$$\begin{aligned}\rho_\epsilon(x) &= \int_{\mathcal{M}} k_\epsilon(x,y)\rho(y)dy \\ &= (2\pi\epsilon)^{d/2}|M|^{1/2}\left(\rho + \epsilon\left[-\nabla\rho^\top\omega_1 - \rho\omega_2 + \frac{1}{2}\mathsf{tr}(M\nabla\nabla\rho)\right] + O(\epsilon^2)\right). \end{aligned} \tag{C.17}$$

Therefore,

$$\rho_\epsilon^{-\alpha} = \frac{(2\pi\epsilon)^{-\alpha d/2}|\tilde{M}|^{\alpha/2}}{\rho^\alpha}\left[1 - \alpha\epsilon\frac{-\nabla\rho^\top\omega_1 - \rho\omega_2 + \frac{1}{2}\mathsf{tr}(M\nabla\nabla\rho)}{\rho} + O(\epsilon^2)\right]. \tag{C.18}$$

To perform right renormalization, we need to multiply the kernel $k_\epsilon(x,y)$ by $\rho_\epsilon^{-\alpha}(y)$. The dependence of $k_\epsilon(x,y)\rho_\epsilon^{-\alpha}(y)$ on y will be shifted to terms of Taylor expansions. As before, let $z = y - x$. First, we expand $|\tilde{M}(y)|^{\alpha/2}$

$$\begin{aligned}|\tilde{M}(x+z)|^{\alpha/2} &= \left[|\tilde{M}(x)| + \nabla|\tilde{M}(x)|^\top z + \frac{1}{2}z^\top\nabla\nabla|\tilde{M}(x)|z + O(\|z\|^3)\right]^{\alpha/2} \\ &= |\tilde{M}(x)|^{\alpha/2}\left[1 + \frac{\nabla|\tilde{M}(x)|^\top z}{|\tilde{M}(x)|} + \frac{z^\top\nabla\nabla|\tilde{M}(x)|z}{2|\tilde{M}(x)|} + O(\|z\|^3)\right]^{\alpha/2} \\ &= |\tilde{M}(x)|^{\alpha/2}\left[1 + \frac{\alpha}{2}\frac{\nabla|\tilde{M}(x)|^\top z}{|\tilde{M}(x)|} + \frac{1}{2}z^\top A_1(x)z + O(\|z\|^3)\right] \end{aligned} \tag{C.19}$$

where

$$A_1(x) = \frac{\alpha\nabla\nabla|\tilde{M}(x)|)}{2|\tilde{M}(x)|} + \frac{\alpha(\alpha-2)\nabla|\tilde{M}(x)|\nabla|\tilde{M}(x)|^\top}{4|\tilde{M}(x)|^2}.$$

Second, we denote the term multiplied by $\alpha\epsilon$ in (C.18) by R:

$$R := \frac{-\nabla\rho^\top\omega_1 - \rho\omega_2 + \frac{1}{2}\mathsf{tr}(M\nabla\nabla\rho)}{\rho}.$$

Now, using (C.19), we start the calculation of $\mathcal{K}_{\epsilon,\alpha}f(x)$:

$$\begin{aligned}\mathcal{K}_{\epsilon,\alpha}f(x) &:= \int_{\mathcal{M}} \frac{k_\epsilon(x,y)}{\rho_\epsilon^\alpha(y)}\rho(y)f(y)dy \\ &= \int_{\mathcal{M}} k_\epsilon(x,y)\left[\rho_\epsilon^{-\alpha}(y)\rho(y)f(y)\right]dy \end{aligned}$$

$$= \left[\frac{|\tilde{M}(x)|}{(2\pi\epsilon)^d}\right]^{\alpha/2} \int_{\mathcal{M}} k_\epsilon(x,y)\left[\rho^{1-\alpha}(y)f(y)\right] \tag{C.20}$$
$$\times \left[1-\alpha\epsilon R(y)+O(\epsilon^2)\right]$$
$$\times \left[1+\frac{\alpha}{2}\frac{\nabla|\tilde{M}(x)|^\top(y-x)}{|\tilde{M}(x)|}+\frac{1}{2}(y-x)^\top A_1(x)(y-x)+O(\|y-x\|^3)\right]dy.$$

To tackle the integral in (C.20), we split it to several integrals each of which we evaluate using Lemma C.2. For brevity, we will omit arguments x in the gradients and Hessians and in the matrix $\tilde{M}$. We continue:

$$\begin{aligned}
&\int_{\mathcal{M}} k_\epsilon(x,y)\left[\rho^{1-\alpha}(y)f(y)\right]\left[1-\alpha\epsilon R(y)+O(\epsilon^2)\right]\\
&\quad\times\left[1+\frac{\alpha}{2}\frac{\nabla|\tilde{M}|^\top z}{|\tilde{M}|}+\frac{1}{2}z^\top A_1 z+O(\|z\|^3)\right]dz\\
=&\mathcal{G}_\epsilon[\rho^{1-\alpha}(y)f(y)](x)-\alpha\epsilon\mathcal{G}_\epsilon[\rho^{1-\alpha}(y)f(y)R(y)](x)\\
+&\frac{\alpha}{2}\mathcal{G}_\epsilon\left[\rho^{1-\alpha}(y)f(y)\frac{\nabla|\tilde{M}|^\top(y-x)}{|\tilde{M}|}\right](x)\\
+&\frac{1}{2}\mathcal{G}_\epsilon\left[\rho^{1-\alpha}(y)f(y)(y-x)^\top A_1(y-x)\right](x)+\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}O(\epsilon^2).
\end{aligned} \tag{C.21}$$

Applying Lemma C.2 we compute the four operators $\mathcal{G}_\epsilon$ in the last equation:

$$\begin{aligned}
\mathcal{G}_\epsilon[\rho^{1-\alpha}f] = \frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}&\left[\rho^{1-\alpha}f\left[1-\epsilon\omega_2(x)\right]\right.\\
&\left.+\epsilon\left\{-\nabla(f\rho^{1-\alpha})^\top\omega_1+\frac{1}{2}\mathsf{tr}(M\nabla\nabla\left[\rho^{1-\alpha}f\right])\right\}+O(\epsilon^2)\right];
\end{aligned} \tag{C.22}$$

$$\alpha\epsilon\mathcal{G}_\epsilon[\rho^{1-\alpha}fR] = \frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}\left[\alpha\epsilon\rho^{1-\alpha}fR+O(\epsilon^2)\right]; \tag{C.23}$$

$$\begin{aligned}
\mathcal{G}_\epsilon\left[\rho^{1-\alpha}(y)f(y)\frac{\nabla|\tilde{M}|^\top(y-x)}{|\tilde{M}|}\right](x) = \frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}&\left\{-\epsilon\frac{\rho^{1-\alpha}f\nabla|\tilde{M}|^\top\omega_1}{|\tilde{M}|}\right.\\
+\frac{\epsilon}{2}\mathsf{tr}&\left.\left(M\nabla_y\nabla_y\left[\rho^{1-\alpha}(y)f(y)\frac{\nabla|\tilde{M}|^\top(y-x)}{|\tilde{M}|}\right]_{y=x}+O(\epsilon^2)\right)\right\}.
\end{aligned} \tag{C.24}$$

Let us calculate the Hessian matrix in (C.24):

$$\begin{aligned}
\nabla_y\nabla_y\left[\rho^{1-\alpha}(y)f(y)\frac{\nabla|\tilde{M}|^\top(y-x)}{|\tilde{M}|}\right]_{y=x} &= \nabla_y\left[\nabla_y\left[\rho^{1-\alpha}(y)f(y)\right]\frac{\nabla|\tilde{M}|^\top(y-x)}{|\tilde{M}|}\right.\\
\left.+\rho^{1-\alpha}(y)f(y)\nabla_y\frac{\nabla|\tilde{M}|^\top(y-x)}{|\tilde{M}|}\right]_{y=x} &= 2\nabla\left[\rho^{1-\alpha}f\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}.
\end{aligned} \tag{C.25}$$

Finally, we compute the operator $\mathcal{G}_\epsilon$ in (C.21):

$$\mathcal{G}_\epsilon\left[\rho^{1-\alpha}(y)f(y)(y-x)^\top A_1(y-x)\right](x) = \frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}\left[\frac{\epsilon}{2}\rho^{1-\alpha}f\mathsf{tr}(MA_1)+O(\epsilon^2)\right]. \tag{C.26}$$

Putting together (C.20)–(C.26) we obtain:

$$\mathcal{K}_{\epsilon,\alpha}f(x) = \left[\frac{|\tilde{M}(x)|}{(2\pi\epsilon)^d}\right]^{\frac{\alpha-1}{2}} \left[\rho^{1-\alpha} f\left[1+\epsilon h(x)\right]\right.$$
$$+\epsilon\left\{-\nabla(f\rho^{1-\alpha})^\top\omega_1 + \frac{1}{2}\mathsf{tr}(M\nabla\nabla\left[\rho^{1-\alpha}f\right]) + \frac{\alpha}{2}\mathsf{tr}\left(M\nabla\left[\rho^{1-\alpha}f\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right)\right\}$$
$$\left.+ O(\epsilon^2)\right], \tag{C.27}$$

where

$$h(x) = -\omega_2(x) - \alpha R(x) - \frac{\alpha}{2}\frac{\nabla|\tilde{M}|^\top\omega_1}{|\tilde{M}|} + \frac{1}{4}\mathsf{tr}(MA_1).$$

To facilitate the calculation, we denote the expression in the curly brackets in (C.27) divided by $q^{1-\alpha}$ by

$$B(\rho,f) := \left[\frac{-\nabla(f\rho^{1-\alpha})^\top\omega_1 + \frac{1}{2}\mathsf{tr}(M\nabla\nabla\left[\rho^{1-\alpha}f\right]) + \frac{\alpha}{2}\mathsf{tr}\left(M\nabla\left[\rho^{1-\alpha}f\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right)}{\rho^{1-\alpha}}\right].$$

Then $\mathcal{K}_{\epsilon,\alpha}f(x)$ can be written as:

$$\mathcal{K}_{\epsilon,\alpha}f(x) = \left[\frac{|\tilde{M}(x)|}{(2\pi\epsilon)^d}\right]^{\frac{\alpha-1}{2}} \rho^{1-\alpha}\left[f\left[1+\epsilon h(x)\right] + \epsilon B(\rho,f) + O(\epsilon^2)\right]. \tag{C.28}$$

Observing that $\rho_{\epsilon,\alpha}(x) = K_{\epsilon,\alpha}1$, i.e., we need to use $f\equiv 1$ to get $\rho_{\epsilon,\alpha}(x)$, we calculate the operator $\mathcal{P}_{\epsilon,\alpha}$:

$$\begin{aligned}\mathcal{P}_{\epsilon,\alpha}f(x) &= \int_{\mathcal{M}} \frac{k_{\epsilon,\alpha}(x,y)}{\rho_{\epsilon,\alpha}(x)} f(y)\rho(y)dy \\ &= \frac{f\left\{1+\epsilon h(x)\right\} + \epsilon B(\rho,f) + O(\epsilon^2)}{1+\epsilon h(x) + \epsilon B(\rho,1) + O(\epsilon^2)}.\end{aligned} \tag{C.29}$$

Expanding $\mathcal{P}_{\epsilon,\alpha}f(x)$ in powers of ϵ we obtain:

$$\begin{aligned}\mathcal{P}_{\epsilon,\alpha}f(x) &= \left[f\left(1+\epsilon h(x)\right) + \epsilon B(\rho,f) + O(\epsilon^2)\right]\left[1-\epsilon h(x) - \epsilon B(\rho,1) + O(\epsilon^2)\right] \\ &= f + \epsilon\left[B(\rho,f) - fB(\rho,1)\right] \\ &= f + \epsilon\frac{-\nabla(f\rho^{1-\alpha})^\top\omega_1 + \frac{1}{2}\mathsf{tr}(\tilde{M}^{-1}\nabla\nabla\left[\rho^{1-\alpha}f\right]) + \frac{\alpha}{2}\mathsf{tr}\left(\tilde{M}^{-1}\nabla\left[\rho^{1-\alpha}f\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right)}{\rho^{1-\alpha}} \\ &\quad - \epsilon f\frac{-\nabla(\rho^{1-\alpha})^\top\omega_1 + \frac{1}{2}\mathsf{tr}(\tilde{M}^{-1}\nabla\nabla\left[\rho^{1-\alpha}\right]) + \frac{\alpha}{2}\mathsf{tr}\left(\tilde{M}^{-1}\nabla\left[\rho^{1-\alpha}\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right)}{\rho^{1-\alpha}}\end{aligned} \tag{C.30}$$

Finally, we compute the operator $\mathcal{L}_{\epsilon,\alpha}$, take the limit $\epsilon\to 0$, and obtain the desired result:

$$\begin{aligned}\lim_{\epsilon\to 0}\frac{\mathcal{P}_{\epsilon,\alpha}f(x) - f(x)}{\epsilon} &= \frac{1}{2}\left(\frac{\mathsf{tr}\left(M\left[\nabla\nabla\left[\rho^{1-\alpha}f\right] - f\nabla\nabla\rho^{1-\alpha}\right]\right)}{\rho^{1-\alpha}}\right) \\ &\quad + \frac{\alpha}{2}\left(\frac{\mathsf{tr}\left(M\left[\nabla\left[\rho^{1-\alpha}f\right] - f\nabla\left[\rho^{1-\alpha}\right]\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right)}{\rho^{1-\alpha}}\right) \\ &\quad - \left(\frac{\left[\nabla(f\rho^{1-\alpha}) - f\nabla(\rho^{1-\alpha})\right]^\top\omega_1\right]}{\rho^{1-\alpha}}\right). \quad \square\end{aligned} \tag{C.31}$$

Appendix D. Proof of Corollary 3.1

Proof. Term 1 in (C.31)**.** First we compute

$$\frac{1}{2}\left(\frac{\mathsf{tr}(M\nabla\nabla\left[\rho^{1/2}f\right])}{\rho^{1/2}} - f\frac{\mathsf{tr}(M\nabla\nabla\left[\rho^{1/2}\right])}{\rho^{1/2}}\right). \tag{D.1}$$

Since ρ is the Gibbs density, we have:

$$\rho = Z^{-1}e^{-\beta F}. \quad \text{Hence} \quad \rho^{1/2} = Z^{-1/2}e^{-\frac{\beta}{2}F}, \quad \nabla\rho^{1/2} = -\left[\frac{\beta}{2}\nabla F\right]\rho^{1/2}.$$

We will use the fact that

$$\nabla\nabla[\rho^{1/2}f] = f\nabla\nabla\rho^{1/2} + [\nabla\rho^{1/2}(\nabla f)^\top + \nabla f(\nabla\rho^{1/2})^\top] + \rho^{1/2}\nabla\nabla f.$$

Applying the property $\mathsf{tr}(AB) = \mathsf{tr}(BA)$ and recalling that M is symmetric, we obtain:

$$\begin{aligned}
\mathsf{tr}[M[\nabla\rho^{1/2}(\nabla f)^\top + \nabla f(\nabla\rho^{1/2})^\top]] =& \mathsf{tr}[M\nabla\rho^{1/2}\nabla f^\top] + \mathsf{tr}[M^\top\nabla f\nabla\rho^{1/2}]\\
=& \mathsf{tr}[M\nabla\rho^{1/2}\nabla f^\top] + \mathsf{tr}[\nabla f(\nabla\rho^{1/2})^\top M^\top]\\
=& \mathsf{tr}[M\nabla\rho^{1/2}\nabla f^\top] + \mathsf{tr}[\nabla f(M\nabla\rho^{1/2})^\top]\\
=& \mathsf{tr}[M\nabla\rho^{1/2}\nabla f^\top] + \mathsf{tr}[(M\nabla\rho^{1/2})\nabla f^\top]\\
=& 2\mathsf{tr}[M\nabla\rho^{1/2}\nabla f^\top]\\
=& 2(M\nabla\rho^{1/2})^\top\nabla f.
\end{aligned}$$

This yields:

$$\frac{1}{2}\left(\frac{\mathsf{tr}(M\nabla\nabla\left[\rho^{1/2}f\right])}{\rho^{1/2}} - f\frac{\mathsf{tr}(M\nabla\nabla\left[\rho^{1/2}\right])}{\rho^{1/2}}\right) = \frac{1}{2}\left(\mathsf{tr}[M\nabla\nabla f] - \beta(M\nabla F)^\top\nabla f\right).$$

Note that this is $\beta/2$ times the generator (45) for the case where the diffusion matrix M is constant.

Term 2 in (C.31)**.** Utilizing the fact that

$$\frac{[\nabla(f\rho^{1/2}) - f\nabla(\rho^{1/2})]}{\rho^{1/2}} = \nabla f,$$

we simplify the second term in (C.31):

$$\frac{\alpha}{2}\left(\frac{\mathsf{tr}\left(M\left[\nabla\left[\rho^{1-\alpha}f\right] - f\nabla\left[\rho^{1-\alpha}\right]\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right)}{\rho^{1-\alpha}}\right) = \frac{\alpha}{2}\left(\mathsf{tr}\left[M\nabla f\right]\frac{\nabla|\tilde{M}|^\top}{|\tilde{M}|}\right).$$

Let us introduce the notation

$$R_k := \tilde{M}^{-1/2}\frac{\partial\tilde{M}}{\partial x_k}\tilde{M}^{-1/2} \tag{D.2}$$

It follows from Jacobi's formula for the derivative of the determinant that

$$\frac{1}{|\tilde{M}|}\frac{\partial|\tilde{M}|}{\partial x_k} = \mathsf{tr}\left(\tilde{M}^{-1}\frac{\partial \tilde{M}}{\partial x_k}\right) \equiv \mathsf{tr}\left(\tilde{M}^{-1/2}\frac{\partial \tilde{M}}{\partial x_k}\tilde{M}^{-1/2}\right) \equiv \mathsf{tr}R_k. \tag{D.3}$$

Hence, we obtain:

$$\frac{1}{4}\mathsf{tr}\left(M\nabla f[\mathsf{tr}R_1,\dots,\mathsf{tr}R_d]\right) = \frac{1}{4}\sum_{k=1}^{d}\sum_{i=1}^{d} M_{ki}\frac{\partial f}{\partial x_i}\mathsf{tr}R_k. \tag{D.4}$$

Term 3 in (C.31)**.** Finally, we compute

$$\frac{[\nabla(f\rho^{1/2}) - f\nabla(\rho^{1/2})]^\top\omega_1}{\rho^{1/2}} = \nabla f^\top\omega_1, \quad \text{where} \tag{D.5}$$

$$\frac{(2\pi\epsilon)^{d/2}}{|\tilde{M}|^{1/2}}\omega_{1,i}(x) = \frac{1}{4\epsilon^2}\int\limits_{\mathcal{B}_{0,\sqrt{\epsilon}}} e^{-\frac{z^\top\tilde{M}z}{2\epsilon}} z_i\left[z^\top[\nabla\tilde{M}(x)z]z\right]dz, \quad 1\le i\le d. \tag{D.6}$$

To compute the integral in (D.6), we do the variable change $t := \epsilon^{-1/2}\tilde{M}^{1/2}z$. Then

$$z_i := \epsilon^{1/2}e_i^\top\tilde{M}^{-1/2}t, \quad \text{where} \quad e_i \quad \text{is the standard unit vector.}$$

The polynomial in the integrand in (D.6) resulting from this change is:

$$\begin{aligned} z_i\left[z^\top[\nabla\tilde{M}(x)z]z\right] &= z_i z^\top\left[\sum_{k=1}^{d}\frac{\partial\tilde{M}}{\partial x_k}z_k\right]z = \sum_{k=1}^{d} z_i z^\top\frac{\partial\tilde{M}}{\partial x_k}zz_k \\ &= \epsilon^2\sum_{k=1}^{d} e_i^\top\tilde{M}^{-1/2}tt^\top\tilde{M}^{-1/2}\frac{\partial\tilde{M}}{\partial x_k}\tilde{M}^{-1/2}tt^\top\tilde{M}^{-1/2}e_k. \end{aligned} \tag{D.7}$$

Using the notation R_k introduced in (D.2) we get:

$$\omega_{1,i}(x) = \frac{1}{4(2\pi)^{d/2}}\sum_{k=1}^{d} e_i^\top\tilde{M}^{-1/2}\left[\int\limits_{\mathbb{R}^d} e^{-\frac{t^2}{2}}tt^\top R_k tt^\top dt\right]\tilde{M}^{-1/2}e_k. \tag{D.8}$$

First we compute the integral in the square brackets in (D.8). This integral is a $d\times d$ matrix, and its entries are:

$$(tt^\top R_k tt^\top)_{ij} = \sum_{l=1}^{d}\sum_{m=1}^{d} t_i t_l[R_k]_{lm}t_m t_j.$$

Case $i=j$**:** Taking into account that in order to produce a nonzero integral, we must have $l=m$ in this case. Hence

$$\begin{aligned} \int\limits_{\mathbb{R}^d} e^{-\frac{t^2}{2}}(tt^\top R_k tt^\top)_{ii}dt &= \int\limits_{\mathbb{R}^d} e^{-\frac{t^2}{2}}\sum_{l=1}^{d}[R_k]_{ll}t_l^2t_i^2dt \\ &= (2\pi)^{d/2}\left(3[R_k]_{ii} + \sum_{l\neq i}R_{ll}\right) = (2\pi)^{d/2}\ \left(2[R_k]_{ii} + \mathsf{tr}R_k\right). \end{aligned} \tag{D.9}$$

Case $i \neq j$: In this case, to produce a nonzero integral, we must have $l = i$ and $m = j$ or the other way around. Hence

$$\int_{\mathbb{R}^d} e^{-\frac{t^2}{2}} (tt^\top R_k tt^\top)_{ii} dt = \int_{\mathbb{R}^d} e^{-\frac{t^2}{2}} \left[[R_k]_{ij} + [R_k]_{ji}\right] t_i^2 t_j^2 dt$$
$$= (2\pi)^{d/2} \left([R_k]_{ij} + [R_k]_{ji}\right) = (2\pi)^{d/2} 2[R_k]_{ij} \tag{D.10}$$

as R_k is symmetric.

Therefore, the integral in the square brackets in (D.8) is

$$(2\pi)^{d/2} \left(2R_k + I\mathsf{tr}R_k\right). \tag{D.11}$$

Plugging this result into (D.8) and recalling (D.2) we obtain:

$$\omega_{1,i}(x) = \frac{1}{4} \sum_{k=1}^{d} e_i^\top \tilde{M}^{-1/2} \left[2R_k + I\mathsf{tr}R_k\right] \tilde{M}^{-1/2} e_k$$
$$= \frac{1}{2} \sum_{k=1}^{d} e_i^\top \tilde{M}^{-1} \frac{\partial \tilde{M}}{\partial x_k} \tilde{M}^{-1} e_k + \frac{1}{4} \sum_{k=1}^{d} e_i^\top \tilde{M}^{-1} \mathsf{tr}R e_k. \tag{D.12}$$

Now we recall that

$$\frac{\partial \tilde{M}^{-1}}{\partial x_k} = -\tilde{M}^{-1} \frac{\partial \tilde{M}}{\partial x_k} \tilde{M}^{-1}.$$

Using this formula, we get:

$$\omega_{1,i}(x) = -\frac{1}{2} \sum_{k=1}^{d} \frac{\partial M_{ik}}{\partial x_k} + \frac{1}{4} \sum_{k=1}^{d} M_{ik} \mathsf{tr}R_k. \tag{D.13}$$

Therefore,

$$\nabla f^\top \omega_1 = -\frac{1}{2} \sum_{i=1}^{d} \sum_{k=1}^{d} \frac{\partial M_{ik}}{\partial x_k} \frac{\partial f}{\partial x_i} + \frac{1}{4} \sum_{i=1}^{d} \sum_{k=1}^{d} M_{ik} \mathsf{tr}R_k \frac{\partial f}{\partial x_i}. \tag{D.14}$$

Getting the final result. Finally, we plug the calculated terms in (C.31). We also use the fact that M is symmetric, i.e. $M_{ik} = M_{ki}$ for $1 \leq i, k \leq d$. We get:

$$\lim_{\epsilon \to 0} L_{\epsilon,\alpha} = \frac{1}{2} \left(\mathsf{tr}[M\nabla\nabla f] - \beta (M\nabla F)^\top \nabla f\right)$$
$$+ \frac{1}{4} \sum_{k=1}^{d} \sum_{i=1}^{d} M_{ki} \frac{\partial f}{\partial x_i} \mathsf{tr}R_k$$
$$+ \frac{1}{2} \sum_{i=1}^{d} \sum_{k=1}^{d} \frac{\partial M_{ik}}{\partial x_k} \frac{\partial f}{\partial x_i} - \frac{1}{4} \sum_{i=1}^{d} \sum_{k=1}^{d} M_{ik} \mathsf{tr}R_k \frac{\partial f}{\partial x_i}$$
$$= \frac{1}{2} \left(\mathsf{tr}[M\nabla\nabla f] - \beta (M\nabla F)^\top \nabla f\right) + \frac{1}{2} (\nabla \cdot M)^\top \nabla f. \tag{D.15}$$

The last expression is the generator $\mathcal{L}$ for the dynamics in collective variables (45) multiplied by $\beta/2$ as desired. □

Appendix E. Obtaining the reactive current and the reaction rate from the committor computed by `mmap`

We have computed the reactive current based on the Gamma operator defined for an Ito diffusion with generator $\mathcal{L}$ as

$$\Gamma(f,g)(x) = \frac{1}{2}\big(\mathcal{L}(fg)(x) - f\mathcal{L}g(x) - g\mathcal{L}f(x)\big). \tag{E.1}$$

This operator is sometimes referred to as the *carré du champ* operator [34,62]. We apply it to the discrete generator matrix L from `mmap`.

Applying the Gamma operator to the generator $\mathcal{L}$ of (4) gives

$$\mathcal{L}(fg) = f\mathcal{L}g + g\mathcal{L}f + 2\beta^{-1}\nabla f^\top M\nabla g, \tag{E.2}$$

and hence $\Gamma(f,g)$ simplifies to

$$\Gamma(f,g)(x) = \beta^{-1}\nabla f^\top M\nabla g(x). \tag{E.3}$$

Choosing $f(x)$ to be the committor $q(x)$ and $g(x)$ to be $\chi_\nu : \mathbb{R}^d \to \mathbb{R}$, mapping x to its νth component, we obtain the νth component of the reactive current by:

$$\rho(x)\Gamma(q,\chi_\nu)(x) = \beta^{-1}\rho(x)[M\nabla q(x)]_\nu. \tag{E.4}$$

Therefore, in order to obtain the reactive current discretized to a dataset, we need to construct a discrete counterpart of the Gamma operator and obtain an estimate for the density ρ.

We recall that the discrete generator L approximates $\frac{\beta}{2}\mathcal{L}$ pointwise on a dataset $\{x_i\}_{i=1}^n$. Let f and g be arbitrary smooth functions and $[f],[g]\in\mathbb{R}^n$ be their discretization to the dataset, i.e., $[f]_i = f(x_i)$ and $[g]_i = g(x_i)$. Since L approximates $\frac{\beta}{2}\mathcal{L}$ pointwise on the dataset, we have:

$$\sum_j L_{ij}([f]_j[g]_j) - [f]_i L_{ij}[g]_j - [g]_i L_{ij}[f]_j \approx \beta\Gamma(f,g)(x_i) = \nabla f^\top M\nabla g(x_i). \tag{E.5}$$

Since the row sums of the matrix L are zeros, the left-hand side of (E.5) can be written as

$$\sum_j L_{ij}([f]_j[g]_j) - [f]_i L_{ij}[g]_j - [g]_i L_{ij}[f]_j = \sum_j L_{ij}([f]_i - [f]_j)([g]_i - [g]_j). \tag{E.6}$$

This allows us to define the discrete analogue of the Gamma operator by:

$$[\hat{\Gamma}(f,g)]_i := \beta^{-1}\sum_{j=1}^n L_{ij}([f]_i - [f]_j)([g]_i - [g]_j). \tag{E.7}$$

Now, it remains to obtain an estimate for the density ρ. We proceed as follows. First we construct an isotropic Gaussian kernel $[\tilde{k}_{\tilde{\epsilon}}]_{ij} = \exp[-||x_i - x_j||^2/(2\tilde{\epsilon})]$. Setting $M(x)\equiv I$ and $f(x)\equiv 1$ in the kernel expansion (C.4), we observe that

$$\lim_{n\to\infty}\frac{1}{n(2\pi\tilde{\epsilon})^{d/2}}\sum_{j=1}^n[\tilde{k}_{\tilde{\epsilon}}]_{ij} = \rho(x) + \mathcal{O}(\tilde{\epsilon}). \tag{E.8}$$

So, we define a kernel density estimate with the vector $[p]\in\mathbb{R}^n$ defined by

$$[p]_i = \frac{1}{n(2\pi\tilde{\epsilon})^{d/2}} \sum_{j=1}^{n} [\tilde{k}_{\tilde{\epsilon}}]_{ij} \qquad i = 1, \dots, n. \tag{E.9}$$

Alternatively, we can use the Mahalanobis kernel from `mmap` with entries $[k_\epsilon]_{ij}$ and utilize (C.4) to define $[p]_i := (n(2\pi\epsilon)^{d/2}|M(x_i)|^{1/2})^{-1} \sum_j [k_\epsilon]_{ij}$ for $i = 1, \dots, n$.

Let $[q] \in \mathbb{R}^n$ be the discrete committor obtained by `mmap`. Using the constructed discrete density $[p]$, we estimate the reactive current using the formula

$$[\hat{\mathcal{J}}]_{\nu i} := \left[p\hat{\Gamma}\left([q], x^\nu\right)\right]_i = \beta^{-1}[p]_i \sum_{j=1}^{n} L_{ij}([q]_i - [q]_j)(x_i^\nu - x_j^\nu) \tag{E.10}$$

where $1 \le \nu \le d$, $1 \le i \le n$, and x_i^ν, x_j^ν denote the ν-th coordinate for data points i and j respectively.

The reaction rate ν_{AB} is given by (9) and can be rewritten as [3]

$$\nu_{AB} = \beta^{-1} \int_{\mathcal{M}\setminus(A\cup B)} \nabla q(x)^\top M(x) \nabla q(x) \rho(x) dx. \tag{E.11}$$

Observing that $\beta^{-1}\nabla q(x)^\top M(x)\nabla q(x) \equiv \Gamma(q,q)$, we get:

$$\nu_{AB} = \int_{\mathcal{M}\setminus(A\cup B)} \Gamma(q,q)(x)\rho(x)dx. \tag{E.12}$$

Hence, we compute an estimate $\hat{\nu}_{AB}$ as the Monte Carlo integral

$$\hat{\nu}_{AB} = \frac{1}{|I_{AB}|} \sum_{i\in I_{AB}} \left[\hat{\Gamma}\left([q],[q]\right)\right]_i = \frac{1}{|I_{AB}|} \sum_{i\in I_{AB}} \sum_{j=1}^{n} L_{ij}([q]_i - [q]_j)^2, \tag{E.13}$$

where $I_{AB} = \{i : x_i \in \mathcal{M}\setminus(A\cup B)\}$.

References

[1] W. E, E. Vanden-Eijnden, Metastability, conformation dynamics, and transition pathways in complex systems, in: Multiscale Modelling and Simulation, Springer, 2004, pp. 35–68.

[2] W. E, E. Vanden-Eijnden, Towards a theory of transition paths, J. Stat. Phys. 123 (3) (2006) 503.

[3] W. E, E. Vanden-Eijnden, Transition-path theory and path-finding algorithms for the study of rare events, Annu. Rev. Phys. Chem. 61 (2010) 391–420.

[4] M. Cameron, Estimation of reactive fluxes in gradient stochastic systems using an analogy with electric circuits, J. Comput. Phys. 247 (2013) 137–152, https://doi.org/10.1016/j.jcp.2013.03.054.

[5] L. Maragliano, A. Fischer, E. Vanden-Eijnden, G. Ciccotti, String method in collective variables: minimum free energy paths and isocommittor surfaces, J. Chem. Phys. 125 (2) (2006) 024106, https://doi.org/10.1063/1.2212942.

[6] P. Ravindra, Z. Smith, P. Tiwary, Automatic mutual information noise omission (AMINO): generating order parameters for molecular systems, Mol. Syst. Des. Eng. 5 (1) (2020) 339–348.

[7] M.A. Rohrdanz, W. Zheng, M. Maggioni, C. Clementi, Determination of reaction coordinates via locally scaled diffusion map, J. Chem. Phys. 134 (12) (2011) 03B624.

[8] S. Piana, A. Laio, Advillin folding takes place on a hypersurface of small dimensionality, Phys. Rev. Lett. 101 (20) (2008) 208101.

[9] F. Sittel, G. Stock, Perspective: identification of collective variables and metastable states of protein dynamics, J. Chem. Phys. 149 (15) (2018) 150901.

[10] Y. Khoo, J. Lu, L. Ying, Solving for high-dimensional committor functions using artificial neural networks, Res. Math. Sci. 6 (1) (2019) 1.

[11] Q. Li, B. Lin, W. Ren, Computing committor functions for the study of rare events using deep learning, J. Chem. Phys. 151 (5) (2019) 054112.

[12] G.M. Rotskoff, E. Vanden-Eijnden, Learning with rare data: using active importance sampling to optimize objectives dominated by rare events, arXiv preprint, arXiv:2008.06334, 2020.

[13] Z. Trstanova, B. Leimkuhler, T. Lelièvre, Local and global perspectives on diffusion maps in the analysis of molecular systems, Proc. R. Soc. A 476 (2233) (2020) 20190036, https://doi.org/10.1098/rspa.2019.0036.
[14] R. Lai, J. Lu, Point cloud discretization of Fokker–Planck operators for committor functions, Multiscale Model. Simul. 16 (2) (2018) 710–726.
[15] R.R. Coifman, S. Lafon, Diffusion maps, Appl. Comput. Harmon. Anal. 21 (1) (2006) 5–30, https://doi.org/10.1016/j.acha.2006.04.006.
[16] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[17] J.B. Tenenbaum, V. De Silva, J.C. Langford, A global geometric framework for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[18] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.
[19] A.L. Ferguson, A.Z. Panagiotopoulos, P.G. Debenedetti, I.G. Kevrekidis, Systematic determination of order parameters for chain dynamics using diffusion maps, Proc. Natl. Acad. Sci. 107 (31) (2010) 13597–13602.
[20] A.L. Ferguson, A.Z. Panagiotopoulos, I.G. Kevrekidis, P.G. Debenedetti, Nonlinear dimensionality reduction in molecular simulation: the diffusion map approach, Chem. Phys. Lett. 509 (1–3) (2011) 1–11.
[21] W. Zheng, M.A. Rohrdanz, M. Maggioni, C. Clementi, Polymer reversal rate calculated via locally scaled diffusion map, J. Chem. Phys. 134 (14) (2011) 144109.
[22] R.R. Coifman, I.G. Kevrekidis, S. Lafon, M. Maggioni, B. Nadler, Diffusion maps, reduction coordinates, and low dimensional representation of stochastic systems, Multiscale Model. Simul. 7 (2) (2008) 842–864.
[23] A.M. Berezhkovskii, A. Szabo, N. Greives, H.-X. Zhou, Multidimensional reaction rate theory with anisotropic diffusion, J. Chem. Phys. 141 (20) (2014) 11B616_1.
[24] A. Singer, R.R. Coifman, Non-linear independent component analysis with diffusion maps, Appl. Comput. Harmon. Anal. 25 (2) (2008) 226–239, https://doi.org/10.1016/j.acha.2007.11.001.
[25] H. Risken, Fokker-Planck equation, in: The Fokker-Planck Equation, Springer, 1996, pp. 63–95.
[26] W. Chen, A.R. Tan, A.L. Ferguson, Collective variable discovery and enhanced sampling using autoencoders: innovations in network architecture and error function design, J. Chem. Phys. 149 (7) (2018) 072312.
[27] H. Stamati, C. Clementi, L.E. Kavraki, Application of nonlinear dimensionality reduction to characterize the conformational landscape of small peptides, Proteins 78 (2) (2010) 223–235.
[28] D. Wang, P. Tiwary, State predictive information bottleneck, J. Chem. Phys. 154 (13) (2021) 134111.
[29] C. Dellago, P.G. Bolhuis, D. Chandler, Efficient transition path sampling: application to Lennard-Jones cluster rearrangements, J. Chem. Phys. 108 (22) (1998) 9236–9245.
[30] D.J. Wales, Discrete path sampling, Mol. Phys. 100 (20) (2002) 3285–3305.
[31] D. Passerone, M. Parrinello, Action-derived molecular dynamics in the study of rare events, Phys. Rev. Lett. 87 (10) (2001) 108302.
[32] B. Peters, Reaction coordinates and mechanistic hypothesis tests, Annu. Rev. Phys. Chem. 67 (2016) 669–690.
[33] P.L. Geissler, C. Dellago, D. Chandler, Kinetic pathways of ion pair dissociation in water, J. Phys. Chem. B 103 (18) (1999) 3706–3710.
[34] G.A. Pavliotis, Stochastic Processes and Applications: Diffusion Processes, the Fokker-Planck and Langevin Equations, vol. 60, Springer, 2014.
[35] P. Metzner, C. Schütte, E. Vanden-Eijnden, Transition path theory for Markov jump processes, Multiscale Model. Simul. 7 (3) (2009) 1192–1219.
[36] M. Cameron, E. Vanden-Eijnden, Flows in complex networks: theory, algorithms, and application to Lennard–Jones cluster rearrangement, J. Stat. Phys. 156 (3) (2014) 427–454.
[37] T. Berry, J. Harlim, Variable bandwidth diffusion kernels, Appl. Comput. Harmon. Anal. 40 (1) (2016) 68–96.
[38] R. Banisch, Z. Trstanova, A. Bittracher, S. Klus, P. Koltai, Diffusion maps tailored to arbitrary non-degenerate Itô processes, Appl. Comput. Harmon. Anal. 48 (1) (2020) 242–265.
[39] T. Berry, T. Sauer, Local kernels and the geometric structure of data, Appl. Comput. Harmon. Anal. 40 (3) (2016) 439–469.
[40] C.J. Dsilva, R. Talmon, C.W. Gear, R.R. Coifman, I.G. Kevrekidis, Data-driven reduction for a class of multiscale fast-slow stochastic dynamical systems, SIAM J. Appl. Dyn. Syst. 15 (3) (2016) 1327–1351, https://doi.org/10.1137/151004896.
[41] R. Talmon, R.R. Coifman, Empirical intrinsic geometry for nonlinear modeling and time series filtering, Proc. Natl. Acad. Sci. 110 (31) (2013) 12535–12540.
[42] C. Moosmüller, F. Dietrich, I.G. Kevrekidis, A geometric approach to the transport of discontinuous densities, SIAM/ASA J. Uncertain. Quantificat. 8 (3) (2020) 1012–1035.
[43] F. Dietrich, M. Kooshkbaghi, E.M. Bollt, I.G. Kevrekidis, Manifold learning for organizing unstructured sets of process observations, Chaos 30 (4) (2020) 043108.
[44] A. Singer, R. Erban, I.G. Kevrekidis, R.R. Coifman, Detecting intrinsic slow variables in stochastic dynamical systems by anisotropic diffusion maps, Proc. Natl. Acad. Sci. 106 (38) (2009) 16090–16095.
[45] A. Schwartz, R. Talmon, Intrinsic isometric manifold learning with application to localization, SIAM J. Imaging Sci. 12 (3) (2019) 1347–1391, https://doi.org/10.1137/18m1198752.
[46] C.J. Dsilva, R. Talmon, N. Rabin, R.R. Coifman, I.G. Kevrekidis, Nonlinear intrinsic variables and state reconstruction in multiscale simulations, J. Chem. Phys. 139 (18) (2013) 11B608_1.
[47] E. Peterfreund, O. Lindenbaum, F. Dietrich, T. Bertalan, M. Gavish, I.G. Kevrekidis, R.R. Coifman, Local conformal autoencoder for standardized data coordinates, Proc. Natl. Acad. Sci. 117 (49) (2020) 30918–30927.
[48] F. Gilani, J. Harlim, Approximating solutions of linear elliptic PDE's on a smooth manifold using local kernel, J. Comput. Phys. 395 (2019) 563–582.
[49] G.J. Moro, F. Cardin, Saddlepoint avoidance due to inhomogeneous friction, Chem. Phys. 235 (1–3) (1998) 189–200.

[50] M.E. Johnson, G. Hummer, Characterization of a dynamic string method for the construction of transition pathways in molecular reactions, J. Phys. Chem. B 116 (29) (2012) 8573–8583.
[51] V.V. Nikulin, I.R. Shafarevich, Geometries and Groups, Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Barcelona, Budapest, 1987.
[52] G.A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, G. Bussi, Plumed 2: new feathers for an old bird, Comput. Phys. Commun. 185 (2) (2014) 604–613, https://doi.org/10.1016/j.cpc.2013.09.018.
[53] D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A.E. Mark, H.J. Berendsen, Gromacs: fast, flexible, and free, J. Comput. Chem. 26 (16) (2005) 1701–1718, https://doi.org/10.1002/jcc.20291.
[54] G.A. Tribello, M. Ceriotti, M. Parrinello, A self-learning algorithm for biased molecular dynamics, Proc. Natl. Acad. Sci. 107 (41) (2010) 17509–17514.
[55] S.-T. Tsai, Z. Smith, P. Tiwary, Reaction coordinates and rate constants for liquid droplet nucleation: quantifying the interplay between driving force and memory, J. Chem. Phys. 151 (15) (2019) 154106.
[56] L. Martini, A. Kells, R. Covino, G. Hummer, N.-V. Buchete, E. Rosta, Variational identification of Markovian transition states, Phys. Rev. X 7 (3) (2017) 031060.
[57] L. Maragliano, E. Vanden-Eijnden, A temperature accelerated method for sampling free energy and determining reaction pathways in rare events simulations, Chem. Phys. Lett. 426 (1–3) (2006) 168–175.
[58] O. Valsson, P. Tiwary, M. Parrinello, Enhancing important fluctuations: rare events and metadynamics from a conceptual viewpoint, Annu. Rev. Phys. Chem. 67 (2016) 159–184.
[59] E. Carter, G. Ciccotti, J.T. Hynes, R. Kapral, Constrained reaction coordinate dynamics for the simulation of rare events, Chem. Phys. Lett. 156 (5) (1989) 472–477.
[60] J.G. Kirkwood, Statistical mechanics of fluid mixtures, J. Chem. Phys. 3 (5) (1935) 300–313.
[61] J.E. Straub, M. Borkovec, B.J. Berne, Calculation of dynamic friction on intramolecular degrees of freedom, J. Phys. Chem. 91 (19) (1987) 4995–4998.
[62] D. Bakry, I. Gentil, M. Ledoux, Analysis and Geometry of Markov Diffusion Operators, vol. 348, Springer Science & Business Media, 2013.