Zuse Institute Berlin

Takustrasse 7
D-14195 Berlin-Dahlem
Germany

Andreas Bittracher[1], Ralf Banisch[1] and
Christof Schütte[1,2]

[1] *Department of Mathematics and Computer Science, Freie Universität Berlin, Germany*

[2] *Zuse Institute Berlin, Germany*

Data-driven Computation of Molecular Reaction Coordinates

ZIB Report 17-77 (December 2017)

Zuse Institute Berlin
Takustrasse 7
D-14195 Berlin-Dahlem

Telefon: 030-84185-0
Telefax: 030-84185-125

e-mail: bibliothek@zib.de
URL: http://www.zib.de

ZIB-Report (Print) ISSN 1438-0064
ZIB-Report (Internet) ISSN 2192-7782

Data-driven Computation of Molecular Reaction Coordinates

Andreas Bittracher[1], Ralf Banisch[1], and Christof Schütte[1,2]

[1]Department of Mathematics and Computer Science, Freie Universität Berlin, Germany
[2]Zuse Institute Berlin, Germany

Abstract

The identification of meaningful reaction coordinates plays a key role in the study of complex molecular systems whose essential dynamics is characterized by rare or slow transition events. In a recent publication, the authors identified a condition under which such reaction coordinates exist – the existence of a so-called transition manifold – and proposed a numerical method for their point-wise computation that relies on short bursts of MD simulations.

This article represents an extension of the method towards practical applicability in computational chemistry. It describes an alternative computational scheme that instead relies on more commonly available types of simulation data, such as single long molecular trajectories, or the push-forward of arbitrary canonically-distributed point clouds. It is based on a Galerkin approximation of the transition manifold reaction coordinates, that can be tuned to individual requirements by the choice of the Galerkin ansatz functions. Moreover, we propose a ready-to-implement variant of the new scheme, that computes data-fitted, mesh-free ansatz functions directly from the available simulation data. The efficacy of the new method is demonstrated on a realistic peptide system.

Introduction

In recent years, it has become possible to numerically explore the chemically relevant slow transition processes in systems with several thousands of atoms. This was made possible due to the increase of raw computational power and deployment of specialized computing architectures [28], as well as by the development of accellerated integration schemes that bias the dynamics in the favor for the slow transition processes, yet preserve the original statistics [1, 11, 17].

To obtain chemical insight about the essential dynamics of the system, this vast amount of high-dimensional data has to be adequately processed and filtered. One desirable goal often is a simplified model of the mechanism of action, in which the fast,

unimportant processes are averaged out or otherwise disregarded. One way is to construct *kinetic* models of the system, i.e. identifying metastable reactant-, product- and possibly intermediate states, and reducing the dynamics to a jump process between them. Under certain regularity assumptions on the root model that are readily fulfilled, such a model can be built in an automated, data-driven fashion [5, 26]. However, the simplicity of the resulting model comes with a price: since the long-time relaxation kinetics is described just by jumps between finitely-many discrete states, any information about the transition process and its dynamical features is lost.

An alternative collection of approaches, to which this paper ultimately contributes, thus aims at the automated identification of good *reaction coordinates* or *order parameters*, mappings from the full to some lower-dimensional, but still *continuous* state space, onto which the full dynamics can be projected without loss of the essential processes. Often enough, this reaction coordinate alone (i.e. without the corresponding dynamical model) already contains more valuable chemical information than the kinetic models, as for example the free energy profile along the reaction coordinate allows the determination of the activation energy of the respective transition process [19].

The systematic and mathematically rigorously motivated construction of reaction coordinates is an area of active research, for an overview see [19]. Where it is available, chemical expert knowledge can be used to guide the construction [7, 31]. In the context of *transition path theory* [32], the committor function is known to be an ideal reaction coordinate [20]. Related to this, approximations to the dominant eigenfunctions of the transfer operator are also often considered ideal reaction coordinates [26, 8, 24], which has been confirmed in [12] for a class of systems with local timescale separation. However, the computation of both committor functions and transfer operator eigenfunctions is infeasible for very high-dimensional systems. Moreover, the authors have recently shown that said eigenfunctions yield redundant reaction coordinates, in the sense that often a further reduction is possible [4].

In the same work, the authors identified necessary characteristics that reaction coordinates have to exhibit in order to retain the slow processes: Their isolines must correspond to those of the dominant transfer operator eigenfunctions. What is more, it was shown that the existence of such reaction coordinates is guaranteed by the existence of a so-called *transition manifold* $\mathbb{M}$, a low-dimensional manifold in the function space L^1. The property that defines $\mathbb{M}$ is that, on moderate time scales $t_{\text{fast}} < t \ll t_{\text{slow}}$, the *transition density functions* of the dynamics concentrate around $\mathbb{M}$. A firm mathematical theory for the existence and identification of reaction coordinates was developed around this transition manifold.

The main practical result of [4] was the insight that any parametrization of $\mathbb{M}$ can be turned into a good reaction coordinate. A numerical algorithm was proposed that allows the point-wise computation of this reaction coordinate and only requires the ability to generate trajectories of the aforementioned moderate length that start at the desired evaluation point.

While the method has a solid theoretical foundation and is directly applicable in many cases, there exists a certain gap to common scenarios in computational practice:

1. While the ability to efficiently compute the reaction coordinate only in specific points is quite remarkable, in practice one often wishes to learn the reaction coordinate in *all* of the accessible state space (i.e. where data is available), as the location of the "interesting" points is unknown in advance.

2. The original method cannot compute the reaction coordinate from dynamical "bulk data" – such as long equilibrated trajectories or the push-forward of point clouds that sample the canonical ensemble – that is preferably generated by contemporary simulation methods and software.

In the present work we attempt to close this gap by proposing an alternative, purely data-driven algorithm for computing the transition manifold reaction coordinate. It is based on a classical Galerkin approximation of the reaction coordinate with an arbitrary ansatz space. As with all Galerkin approximations, its numerical approximation requires only a so-called transition matrix between its discretisation elements. This matrix can be constructed for example from the aforementioned types of bulk data. We will see that the same transition matrix appears as an essential ingredient in the construction of the aforementioned Markov state models. This makes our new method applicable whenever the construction of such a discrete kinetic model is possible.

Finally, with the objective to create a method that requires only a minimum of a priori information about the system, we propose a very practical implementation of this Galerkin approximation that constructs a mesh-free set of Voronoi cell-based ansatz functions directly from the available simulation data. Interestingly, the task of optimally choosing the Voronoi centers leads to two well-known and highly scalable algorithms from data mining, namely the k-Means clustering algorithm and Poisson-disk sampling algorithm, depending on the chosen error measure. We demonstrate the efficacy of this method in identifying chemically interpretable essential degrees of freedom and retaining the original slow timescales of the small peptide Dialanine.

The paper is organized as follows: In Section 1 the basic concepts of timescale-separated processes and reaction coordinates are introduced. Section 2 reviews the established transition manifold theory as well as the local burst-based algorithm. In Section 3, the Galerkin approximation of the transition manifold reaction coordinate is derived as well as the Voronoi-based implementation. Section 4 demonstrates the application of our new method to a simple synthetic example system, as well as to the realistic Dialanine system.

1. Timescale-separated systems and reaction coordinates

We model our molecular dynamical system as a time-homogenous, stationary and ergodic stochastic process $\{X_t\}_{t\geq 0}$ on a state space $\mathbb{X} \subset \mathbb{R}^n$. Given a starting point $x \in \mathbb{X}$, the further stochastic evolution of X_t can be described by the time-parametrized family of *transition density functions* $p_x^t : \mathbb{X} \to \mathbb{R}_{\geq 0}$, which is defined by

$$\mathsf{P}[X_t \in \mathbb{A} \mid X_0 = x] = \int_{\mathbb{A}} p_x^t(y)\ \mathrm{d}y \quad \forall \text{ measurable } \mathbb{A} \subset \mathbb{X}\ .$$

Here $\mathsf{P}[\cdot|\cdot]$ denotes the conditional probability. $p_x^t(y)$ can thus be interpreted as the conditional density of $X_t = y$ provided that $X_0 = x$.

Stationarity together with ergodicity implies the existence of a unique equilibrium density ρ that is invariant under X_t, i.e. that fulfills

$$X_0 \sim \rho \Rightarrow X_t \sim \rho \quad \forall t \geq 0 \ .$$

Let μ denote the associated invariant measure, given by $\mathrm{d}\mu = \rho \ \mathrm{d}x$.

As the only further technical assumption we require that X_t is *reversible*, i.e. that

$$\rho(x) \ p_x^t(y) = \rho(y) \ p_y^t(x)$$

holds for all $x, y \in \mathbb{X}$. Again, this condition is usually satisfied by common molecular models.

Transfer operators and implied timescales. To describe and distinguish "sub-processes" or "events" within X_t with different equilibration times, the *transfer operator* $\mathcal{T}^t : L^1_\mu \to L^1_\mu$ and its eigenpairs will take a central role. It describes the time-evolution of X_t with an arbitrary portion of ρ as the starting density, i.e.

$$X_0 \sim u \cdot \rho \quad \Rightarrow \quad X_t \sim \big(\mathcal{T}^t u\big)\rho \ .$$

Formally it is defined as

$$\mathcal{T}^t u(x) = \int_{\mathbb{X}} \frac{\rho(y)}{\rho(x)} p_x^t(y) u(y) \ \mathrm{d}y.$$

ρ being the unique invariant density directly implies that the constant function $v_0(x) \equiv 1$ is the only eigenfunction of $\mathcal{T}^t$ to eigenvalue $\lambda_0 = 1$. Furthermore, it can be shown that $\mathcal{T}^t$ is self-adjoint and contractive under an appropriate expansion onto L^2_μ [2], and as such possesses a real positive point spectrum

$$1 = \lambda_0^t < \lambda_1^t \leq \ldots \leq \lambda_m^t$$

with the remaining spectrum being confined to a ball around the origin of radius $R^t \leq \lambda_m^t$.

It is well known [] that the eigenpair $(\lambda_i^t, v_i), \ i = 1, \ldots, m$ then correspond to the i-th slowest-decaying sub-processes of X_t, with the *implied timescale* of this process being defined by

$$t_i = -t / \log(\lambda_i^t) \ . \tag{1}$$

We are interested in the case where there exist a $1 \leq d < m$ such that the d slowest processes are well-separated from the rest by a *timescale gap*, i.e. $t_1 - t_d \ll t_d - t_{d+1}$. The reproduction of these d slowest processes and the associated timescales will be the primary objective of our coarse graining approaches.

Characterization of good reaction coordinates. In our workflow, the first step of coarse graining a MD system is the identification of a good *reaction coordinate* (RC). Following the notions of [18], a RC is simply any C^1 function $\xi : \mathbb{X} \to \mathbb{R}^k$, i.e. an observable of the full system's state. One may think of a map from the atom's cartesian positions to some internal degrees of freedom, e.g. certain dihedral angles or interatomic distances.

The now k-dimensional *projected process* is then given by $\xi(X_t)$. As mentioned above, the the quality of ξ is determined by how well the slowest timescales of $\xi(X_t)$ correspond to those of the full process X_t. For that, we consider the *projected transfer operator* associated with $\xi(X_t)$, which is given by $\mathcal{T}_\xi^t : L^1_\mu \to L^1_\mu$,

$$\mathcal{T}_\xi^t = P_\xi \mathcal{T}^t P_\xi \ ,$$

with the *coordinate projection*

$$P_\xi f(x) = \mathsf{E}_\mu[f(Y) \mid \xi(Y) = \xi(x)] \ .$$

In other words, P_ξ takes the μ-weighted average of f along the $\xi(x)$-level set of ξ and assigns it to each point of the level set[1]. Analogous to $\mathcal{T}^t$, $\mathcal{T}_\xi^t$ describes the transport of L^1_μ-densities of the process $\xi(X_t)$.

Due to the definition (1) of implied timescales by the eigenvalues of $\mathcal{T}^t$, we now vaguely call ξ a "good" reaction coordinate, if

$$\sigma_{\mathrm{dom}}(\mathcal{T}^t) \approx \sigma_{\mathrm{dom}}(\mathcal{T}_\xi^t) \ , \tag{2}$$

where σ_{dom} is the set of the $d+1$ leading eigenvalues of the respective operator.

In [4] it has been shown that (2) is fulfilled in an appropriate sense whenever ξ *parametrizes* the associated eigenfunctions:

Theorem 1.1 (Corollary 3.6 and Lemma 4.2 in [4])**.** *Let (λ_i^t, v_i) be an eigenpair of $\mathcal{T}^t$. Assume there exists a function $\tilde{v}_i : \mathbb{R}^k \to \mathbb{R}$ with*

$$|v_i(x) - \tilde{v}_i(\xi(x))| \leq \varepsilon \quad \forall x \in \mathbb{X} \ . \tag{3}$$

Then there exists an eigenvalue $\tilde{\lambda}_i^t$ of $\mathcal{T}_\xi^t$ with $|\lambda_i^t - \tilde{\lambda}_i^t| \leq \frac{2\varepsilon}{\sqrt{1-\varepsilon^2}}$.

Intuitively, ξ is a good RC if its isolines run almost parallel to the isolines of all the dominant eigenfunctions v_i. Thus, naturally, with the collection of the d dominant eigenfunctions as RC, $\xi := (v_1, \ldots, v_d)^\intercal$ the eigenvalue error (2) is zero. However, working with dominant eigenfunctions as RCs has two major disadvantages:

1. The eigenproblem is inherently global. If we we wish to learn the value of an eigenfunction v_i at only one location $x \in \mathbb{X}$, we need an approximation of $\mathcal{T}^t$ that is is accurate on all of $\mathbb{X}$. However, the computational effort to construct such an approximation grows exponentially[2] with $\dim(\mathbb{X})$.

[1] Note that $P_\xi f$ is itself a function on $\mathbb{X}$, but is constant along each level set of ξ, and thus essentially depends only on $\xi(\mathbb{X}) \subset \mathbb{R}^k$.

[2] There have been attempts to mitigate this ([33, 15]), but we aim to circumvent this problem entirely.

2. In metastable systems, the number of dominant eigenfunctions $(d+1)$ equals the number of metastable states. This number can be much larger than the optimal dimension of a good RC. Thus, in general, a RC comprised of the dominant eigenfunctions is redundant.

2. The transition manifold

In [4] it was shown how to find such RCs under the assumption that the set of probability densities $\{p_x^t \mid x \in \mathbb{X}\}$, "cluster" around a low-dimensional manifold in L^1 for the right choice of lag time t:

Assumption 2.1. There exists a lag time t and an r-dimensional manifold $\mathbb{M} \subset L^1(\mathbb{X})$ so that for every $x \in \mathbb{X}$,

$$\min_{f \in \mathbb{M}} \|f - p_x^t\| \leq \varepsilon \ .$$

We call such an $\mathbb{M}$ a *transition manifold*. It can be argued that this assumption is naturally fulfilled in systems with metastable sets that are connected by a network of transition pathways. For a brief illustration, consider a system with two metastable sets that are connected by a one-dimensional transition pathway (Figure 1). Here, we choose the lag time t to fall in between the fast and slow time scales: $t_{\text{fast}} \ll t \ll t_{\text{slow}}$. We can assume that, after this intermediate lag time, the fast parts of the dynamics are equilibrated, which means that a typical realization of X_t has left the transition region (if it started there) and moved to one of the metastable sets, has significantly sampled the associated quasistationary density, but has not had enough time to sample the slow transition process between the metastable states. Consequently, p_x^t is essentially a convex combination of the quasistationary densities, with coefficients representing the probability to hit the corresponding metastable set from x within time t. This probability, however, does depend only on the progress of x along the transition pathway, and thus p_x^t (as a function in x) is almost constant on certain fibers perpendicular to the transition pathway. As the transition pathway is one-dimensional, the p_x^t almost lie on a one-dimensional manifold in L^1.

Remark 2.2. This argument has been made somewhat more precise in [4]. However, rigorous conditions for the existence of a transition manifold of certain dimension are still lacking and subject of current investigation. In particular, it is easy to find examples that show that metastability alone is neither sufficient nor required for a transition manifold to exist.

Assumption 2.1 guarantees the existence of prototypical *ideal* reaction coordinates, that however prove *uncomputable* in practice: Let $\mathcal{Q} : L^1 \to \mathbb{M}$ be the orthogonal projection onto the transition manifold, i.e.

$$\mathcal{Q}(f) := \arg\min_{g \in \mathbb{M}} \|f - g\|,$$

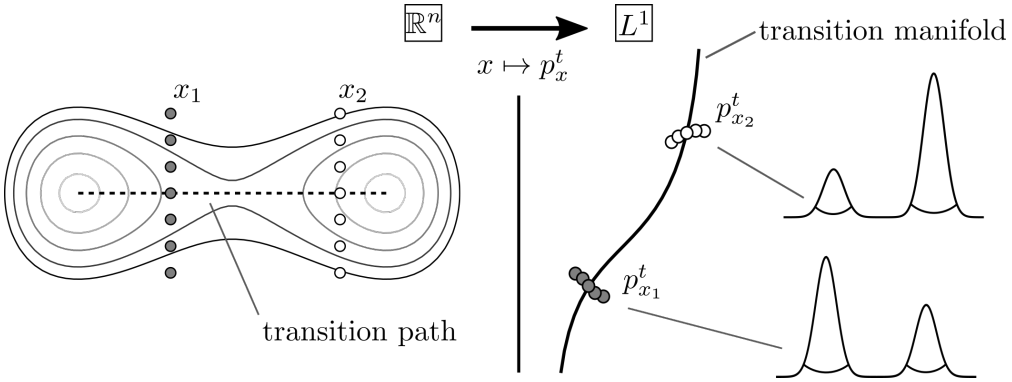


Figure 1

and let $\mathcal{E} : \mathbb{M} \to \mathbb{R}^k$ be an embedding of $\mathbb{M}$[3]. Then one can show [4] that the reaction coordinate

$$\xi(x) := \big(\mathcal{E} \circ \mathcal{Q}\big)(x) \tag{4}$$

fulfills the requirement of Theorem 1.1 and thus guarantees a good approximation of the dominant timescales.

Local computation of reaction coordinates. Unfortunately, the transition manifold $\mathbb{M}$ and thus both the projection $\mathcal{Q}$ and the embedding $\mathcal{E}$ are unknown. In practice, both maps have to be approximated.

- For $\mathcal{Q}$, we see from Assumption 2.1 that $\|\mathcal{Q}(x) - p_x^t\| \leq \varepsilon$ and replace the map $x \mapsto \mathcal{Q}(x)$ by $x \mapsto p_x^t$.
- For $\mathcal{E}$, we call on an infinite-dimensional variant of the weak Whitney embedding theorem [14] to see that *almost every*[4] linear map $\mathcal{F} \to \mathbb{R}^k$ is an embedding of the r-dimensional manifold $\mathbb{M}$, if only $k \geq 2r + 1$. The map $\mathcal{E}$ can thus be replaced by an *arbitrarily chosen* linear map $\mathcal{F} : L^1 \to \mathbb{R}^{2r+1}$.

Choosing the components embedding $\mathcal{F}$ to be of the form

$$\mathcal{F}_i(f) := \langle f, \eta_i \rangle = \int f \eta_i \ , \quad i = 1, \ldots, 2r + 1 \ ,$$

with arbitrarily chosen observables $\eta_i \in L^\infty$ then finally leads to the *transition manifold reaction coordinate* (TMRC)

$$\tilde{\xi}(x) := \mathbb{E}[\eta(X_t) \mid X_0 = x] \ . \tag{5}$$

As a sidenote, this is simply the Koopman operator [16, 21] applied to η: $\tilde{\xi}(x) = \mathcal{K}^t \eta(x)$

To compute the value of $\tilde{\xi}$ at specific evaluation points $\mathbb{X}_N := \{x_1, \ldots, x_N\} x \subset \mathbb{X}$, an algorithm based on short simulation burst was proposed in [4]. For completeness' sake, it has been added to Appendix A.

[3] By the strong Whitney embedding theorem, there always exists an embedding of dimension $k \leq 2r$.
[4] in the sense of prevalence

3. Galerkin approach for computing reaction coordinates

The above burst-based approach for computing the reaction coordinate has two major disadvantages:

1. $\tilde{\xi}$ can only be computed *pointwise* and has no closed analytic form. For every new evaluation point many numerical MD simulations have to be started. Further, the evaluation points have to be chosen in regions relevant to the slow transition processes (i.e. in the transition regions and metastable sets), which is a non-trivial task.

2. The computation of $\tilde{\xi}$ is based on multiple short, instead of one long MD simulation. Although this can also be seen as an advantage, the way modern MD software works often favors the simulation of single long trajectories. Further, there is a vast archive of already pre-computed trajectories for many interesting metastable systems. If one could compute $\tilde{\xi}$ based on those, those systems could be coarse-grained with minimal effort.

In the following we will describe how a Galerkin approximation to ξ can be computed from a wider range of simulation data. It is by construction analytically defined on the whole region where data is available and can be evaluated cheaply.

3.1. Galerkin approximation of reaction coordinates

To avoid an overload of subscripts, we from now on understand scalar products, integrals and application of operators element-wise, e.g. with $\tilde{\xi} : \mathbb{R}^n \to \mathbb{R}^{2r+1}$ and $\varphi_k : \mathbb{R}^n \to \mathbb{R}$,

$$\langle \tilde{\xi}, \varphi_k \rangle_\mu = \big(\langle \tilde{\xi}_1, \varphi_k \rangle_\mu, \dots, \langle \tilde{\xi}_{2r+1}, \varphi_k \rangle_\mu \big)^\mathsf{T}$$

Further assume from now on that $\eta \in L^\infty \cap L^2_\mu$ (this implies $\tilde{\xi} \in L^\infty \cap L^2_\mu$).

Chose an ansatz space $\mathcal{V}_N \subset L^\infty \cap L^2_\mu$ with basis $\{\varphi_1, \dots, \varphi_N\}$. The Galerkin projection [**?**] of $\tilde{\xi}$ then is defined as

$$\Pi_N \tilde{\xi} = \sum_{k,j=1}^{N} (S^{-1})_{kj} \langle \varphi_k, \tilde{\xi} \rangle_\mu \varphi_j \ , \tag{6}$$

with the nonnegative, symmetric *weight matrix* $S \in \mathbb{R}^{N \times N}$

$$S_{kj} = \langle \varphi_k, \varphi_j \rangle_\mu \ .$$

For many popular choices of basis functions, this matrix is given analytically or can be cheaply computed.

Recall that $\tilde{\xi}$ can also be written in terms of the Koopman operator: $\tilde{\xi}(x) = \mathcal{K}^t \eta(x)$. Let $\eta_N \in L^\infty \cap \mathcal{V}_N$, $\eta_N = \sum_{l=1}^N c_l \varphi_l$ be an observable with $\|\eta - \eta_N\|_{L^2_\mu} \leq \varepsilon$. Then

$$\|\mathcal{K}^t \eta - \mathcal{K}^t \eta_N\|_{L^2_\mu} \leq \underbrace{\|\mathcal{K}^t\|_{L^2_\mu}}_{=1} \|\eta - \eta_N\|_{L^2_\mu} \leq \varepsilon \ .$$

Here it was used that $\mathcal{K}^t$ can be considered a non-expanding operator on L^2_μ, see [2].

With this, the scalar product in (6) can be estimated as follows:

$$\langle \varphi_k, \tilde{\xi} \rangle_\mu = \langle \varphi_k, \mathcal{K}^t \eta \rangle_\mu \approx \langle \varphi_k, \mathcal{K}^t \eta_N \rangle_\mu = \sum_{l=1}^{N} \langle \varphi_k, \mathcal{K}^t \varphi_l \rangle_\mu c_l \ . \tag{7}$$

Thus, in order to estimate (6), all that has to be computed are the entries of the *transition matrix* $T \in \mathbb{R}^{N \times N}$,

$$T_{kl} = \langle \varphi_k, \mathcal{K}^t \varphi_l \rangle_\mu \ .$$

Remark 3.1. This is the same transition matrix on which the Galerkin approximation of the transfer operator eigenfunctions, and thus such popular methods as Markov State Model (MSM) analysis are based [27, 10]. The Galerkin approximation of $\tilde{\xi}$ is thus applicable whenever those methods are.

Data-based computation of the transition matrix. The entries of T can now be approximated based on simulation data. Let $\hat{\mathbb{X}} \subset \mathbb{X}$ be the region of interest in which the the computation of the reaction coordinate is desired, and let $\mathcal{V} \subset L^2_\mu(\mathbb{X})$ be a basis that adequately resolves $L^2_\mu(\hat{\mathbb{X}})$.

The approximation of T requires two sets of data points,

$$\mathbb{X}_M = \{x_0, \ldots, x_M\} \ , \quad \mathbb{Y}_M = \{y_0, \ldots, y_M\} \ , \quad \mathbb{X}_M, \mathbb{Y}_M \subset \mathbb{X} \ ,$$

where $\mathbb{X}_M$ samples the stationary measure μ, and $\mathbb{Y}_M$ contains realizations of the dynamics with starting points $\mathbb{X}_M$, $y_i = \Phi^t x_i$, with t the lag time from Assumption 2.1, and Φ^t denoting the random variable that maps the initial state of a (stochastic) MD trajectory to its state after time t. This data can for example be obtained from a single equilibrated numerical trajectory of step size τ,

$$\mathbb{X}_M = \{x_0, \Phi^\tau x_0, \ldots, \Phi^{(M-1)t} x_0\} \ , \quad \mathbb{Y}_M = \{\Phi^t x_0, \Phi^{\tau+t} x_0, \ldots, \Phi^{(M-1)\tau+t} x_0\} \ , \tag{8}$$

or the concatenation of multiple trajectories that *together* sufficiently sample μ. Alternatively, $\mathbb{X}_M$ could be the output of an enhanced sampling algorithm, such as Markov chain Monte Carlo methods [13], and $\mathbb{Y}_M$ its time-t push forward.

We then get for the entries of T

$$T_{kl} = \int_{\mathbb{X}} \varphi_k(x) \mathcal{K}^t \varphi_l(x) \, d\mu(x) \ ,$$

which can be approximated by a Monte Carlo sum with μ-distributed sampling points. As the points in $\mathbb{X}_M$ are μ-distributed, this becomes

$$\approx \frac{1}{M} \sum_{j=0}^{M} \varphi_k(x_j) \mathcal{K}^t \varphi_l(x_j) = \frac{1}{M} \sum_{j=0}^{M} \varphi_k(x_j) \mathbb{E}[\varphi_l(X_t) \mid X_0 = x_j] \ .$$

If the number of sampling points M is sufficiently high, the expectation value can be approximated by a single realization.We then get

$$T_{kl} \approx \frac{1}{M} \sum_{j=0}^{M} \varphi_k(x_j)\varphi_l(\Phi^t x_j) = \frac{1}{M} \sum_{j=0}^{M} \varphi_k(x_j)\varphi_l(y_j) \ . \tag{9}$$

Once the T_{kl} are computed, $\Pi_N \tilde{\xi}$ can be evaluated at arbitrary points in $\hat{\mathbb{X}}$ as follows:

$$\Pi_N \tilde{\xi}(x) = \sum_{j,k=1}^{N} \varphi_j(x)(S^{-1})_{jk} \Big(\sum_{l=1}^{N} T_{kl} c_l \Big) \tag{10}$$

3.2. Implementation: Voronoi-based Galerkin approximation

Equation (10) can be turned into an efficient method for approximating $\tilde{\xi}$ for a wide range of suitable chosen ansatz spaces $\mathcal{V}_N$. In this section, we detail a simple, yet practical algorithm that constructs a meshfree ansatz space directly from the available simulation data.

Let $\{A_1, \dots, A_N\}$ be a partition of $\hat{\mathbb{X}}$, i.e. $\bigcup A_i = \hat{\mathbb{X}},\ A_i \cap A_j = \varnothing\ \forall i \neq j$. Choosing the indicator functions $\varphi_k = \mathbf{1}_{A_k}$ as the basis of $\mathcal{V}$, (9) becomes

$$T_{kl} \approx \frac{\#\big\{i \in \{0, \dots, M-m\} \ \big|\ x_i \in A_k \wedge y_i \in A_l\big\}}{\#\big\{i \in \{0, \dots, M-m\} \ \big|\ x_i \in A_k\big\}} \ ,$$

which is effectively just counting the number of transitions from set A_k to set A_l within the data sets $\mathbb{X}_M,\ \mathbb{Y}_M$.

In the case of indicator functions, with $x \in A_k$, we have, with $S_{kk} = \langle \mathbf{1}_{A_k}, \mathbf{1}_{A_k} \rangle_\mu$,

$$\Pi_N \tilde{\xi}(x) = S_{kk}^{-1} \sum_{l=1}^{N} T_{kl} c_l \ .$$

Choosing the sets A_k naively, for example on a regular grid, invokes the curse of dimension. We thus propose a partition into Voronoi cells $\{A_1, \dots, A_N\}$ with suitable center points $\{e_1, \dots, e_N\} \subset \mathbb{X}$ for Galerkin approximation. With this, we will also be able to avoid the explicit construction of the transition matrix.

Commonly, one wishes to approximate $\tilde{\xi}$ in the region $\hat{\mathbb{X}}$ of state space that is covered with the available data points $\mathbb{X}_M$. The question is then how the Voronoi centers $\{e_1, \dots, e_N\} \subset \mathbb{X}$ should be chosen in order to achieve this. In the following, we demonstrate that two different criteria on the approximation quality of $\tilde{\xi}$ lead to two different algorithms for selecting the Voronoi centers.

Minimizing L^2 error. Since $\tilde{\xi} \in L^2_\mu(\hat{\mathbb{X}})$, we may ask to minimize the L^2 error:

$$\|\tilde{\xi} - \tilde{\xi}_N\|_{L^2_\mu(\hat{\mathbb{X}})} \overset{!}{=} \min_{\{e_1, \dots, e_N\} \subset \mathbb{X}} \ . \tag{11}$$

The difficulty is that neither $\tilde{\xi}$ nor $\tilde{\xi}_N$ are known in advance. In the appendix we show that an unbiased estimator of $\|\tilde{\xi} - \tilde{\xi}_N\|^2_{L^2_\mu(\hat{\mathbb{X}})}$ based on the sampled data $\mathbb{X}_M = \{x_1, \ldots, x_M\}$ is given by

$$S_\xi(A_1, \ldots, A_N) = \sum_{k=1}^{N} \sum_{x_i \in A_k} \|\tilde{\xi}(x_i) - \bar{\xi}_{A_k}\|^2_{\mathbb{R}^{2k+1}}, \tag{12}$$

which is the objective function of k-means clustering in the image space of the reaction coordinate $\tilde{\xi}$. Here $\bar{\xi}_{A_k} = |A_k|^{-1} \sum_{x_i \in A_k} \tilde{\xi}(x_i)$ is the mean of $\tilde{\xi}$ in cell A_k. To minimize this objective function directly one would have to know $\tilde{\xi}$. If we assume in addition that $\tilde{\xi}$ is Lipschitz continuous with Lipschitz constant L, then

$$S_\xi(A_1, \ldots, A_N) \leq L^2 \sum_{k=1}^{N} \sum_{x_i \in A_k} \|\tilde{x}_i - e_k\|^2$$

where e_k is such that $\tilde{\xi}(e_k) = \bar{\xi}_{A_k}$. Minimizing this upper bound is achieved by k-means clustering in configurational space. In summary, minimizing the L^2 error leads to k-means clustering in configurational space with $k = N$ as an algorithm to choose the Voronoi centers. The cluster centers returned by k-means are the Voronoi centers $\{e_1, \ldots, e_N\}$, and points are assigned to to the cell with the closest center point. k-means is highly scalable for both large amounts of clusters N and a large number of data points M, and is readily available in most software packages.

Minimizing uniform error. Thinking of $\tilde{\xi}$ as an observable, it is natural to minimize the *uniform* observable error

$$\|\tilde{\xi} - \tilde{\xi}_N\|_{L^\infty(\hat{\mathbb{X}})} \stackrel{!}{=} \min_{\{e_1, \ldots, e_N\} \subset \mathbb{X}} . \tag{13}$$

In the appendix we show that if $\tilde{\xi}$ is Lipschitz continuous with Lipschitz constant L, then

$$\|\tilde{\xi} - \tilde{\xi}_N\|_{L^\infty(\hat{\mathbb{X}})} \leq L \max_{i=1\ldots N} \mathrm{diam}(A_i)$$

where $\mathrm{diam}(A_i)$ is the diameter of the Voronoi cell A_i. Minimizing the upper bound then means looking for Voronoi centers such that the diameter of the largest Voronoi cell is minimized. Since the number of Voronoi cells and the volume of the set $\mathbb{X}_M$ to be covered are fixed, the minimum is achieved if the centers cover $\mathbb{X}_M$ evenly, such that the Voronoi cells all have similar diameters. Therefore, we may alternatively maximize the diameter of the smallest Voronoi cell, which is lower bounded by the minimal internal point distance:

$$\min_{i=1\ldots N} \mathrm{diam}(A_i) \geq \min_{\substack{i,j=1,\ldots,N \\ i \neq j}} \|e_i - e_j\|.$$

The inequality holds because $\min \|e_i - e_j\|$ is twice the distance from e_i to that face of A_i which is closest to e_i, while the diameter of A_i is by definition larger. Maximizing

the lower bound then leads to the objective function of maximal minimal internal point distance:

$$E = \operatorname*{arg\,max}_{\{e_1,\dots,e_N\}\subset\mathbb{X}_M} \min_{\substack{i,j=1,\dots,N\\ i\neq j}} \|e_i - e_j\| \ .$$

This problem is related to *Poisson disk-* or *blue noise (sub)sampling* in computer vision [6]. A simple heuristic *picking algorithm* to compute a set that approximates the above distance is the following [34]:

Algorithm 3.1 Picking algorithm

Input: $\mathbb{X}_M$, N
Output: $E = \{e_1, \dots, e_N\}$
1: $e_1 \leftarrow$ random point from $\mathbb{X}_M$
2: **for** $j = 2, \dots, N$ **do**
3: pick the point with the maximum distance from the previous points:
$e_j \leftarrow \operatorname*{arg\,max}_{x\in\mathbb{X}_M} \min_{i=1,\dots,j-1} \|x - e_i\|$
4: **end for**

In summary, minimizing the L^2 error of $\tilde{\xi}$ leads to k-means clustering as an algorithm for picking the Voronoi centers while minimizing the uniform error of $\tilde{\xi}$ leads to the farthest point picking algorithm 3.1. In section 4 we compare both alternatives. In general k-means will lead to denser Voronoi cells in regions of large μ, while algorithm 3.1 will lead to evenly sized Voronoi cells.

Voronoi-Galerkin approximation. In the following we assume that the Voronoi centers have been chosen to minimize either of the errors (11) or (13), i.e. as the outcome of the k-means algorithm or Algorithm 3.1 applied to our data. With this, consider the ansatz space of characteristic functions over the Voronoi cells of E:

$$\varphi_k(x) := \begin{cases} 1, & e_k = \operatorname*{arg\,min}_{e_j\in E} \|e_j - x\| \\ 0, & \text{otherwise} \end{cases} \ .$$

The corresponding weight matrix is diagonal with $S_{ll} = \langle \varphi_l, \varphi_l \rangle_\mu$. Thus, the l-th Galerkin coefficient of (6) is

$$\frac{\langle \tilde{\xi}, \varphi_l \rangle_\mu}{\langle \varphi_l, \varphi_l \rangle_\mu} = \frac{\int_{\mathbb{X}} \tilde{\xi}(x)\varphi_l(x)\ d\mu(x)}{\int_{\mathbb{X}} \varphi_l(x)^2\ d\mu(x)} \ .$$

Both integrals can again be approximated by a Monte Carlo sum with μ-distributed sampling points, for which again the data set $\mathbb{X}_M$ can be used. Thus we have

$$\begin{aligned} \int_{\mathbb{X}} \tilde{\xi}(x)\varphi_l(x)\ d\mu(x) &\approx \frac{1}{M}\sum_{j=1}^{M} \tilde{\xi}(x_j)\varphi_l(x_j) \approx \sum_{j=1}^{M} \eta(y_j)\varphi_l(x_j) \\ &= \frac{1}{M}\sum \left\{ \eta(y_j) \;\middle|\; \operatorname*{arg\,min}_{i=1,\dots,N} \|e_i - x_j\| = l \right\} \ . \end{aligned}$$

This can be computed simply by assigning the points in $\mathbb{X}_M$ to the Voronoi centers, evaluating the observable η at the corresponding points in $\mathbb{Y}_M$ and summing up.

The other integral is

$$\begin{aligned}\int_{\mathbb{X}} \varphi_l(x)^2 \, d\mu(x) &\approx \frac{1}{M} \sum_{j=1}^{M} \varphi_l(x_j) \\ &= \frac{1}{M} \#\big\{ j \mid \operatorname*{arg\,min}_{i=1,\dots,N} \|e_i - x_j\| = l \big\} \ .\end{aligned}$$

This is just the relative number of data points in $\mathbb{X}_M$ that belong to the Voronoi center e_l.

Overall, to estimate the Galerkin coefficients and thus a functional form of $\tilde{\xi}_N$, we propose the following algorithm:

Algorithm 3.2 Voronoi-Galerkin-approximation of the TMRC

Input: data sets $\mathbb{X}_M$, $\mathbb{Y}_M$, TM dimension r

Output: Galerkin approximation $\tilde{\xi}_N$

1: $\{e_1, \dots, e_N\} \leftarrow$ Voronoi centers, output of either k-means or Algorithm 3.1 applied to $\mathbb{X}_M$
2: $\eta_N \leftarrow$ generic observable with coefficients $\eta_i \in L^\infty(\mathbb{X}) \cap \mathcal{V}_N, \ i = 1, \dots 2r+1$
3: **for** each data point $x_j \in \mathbb{X}_M$ **do**
4: $\quad \mathcal{I}(j) \leftarrow \arg\min_{i=1,\dots,N} \|x_j - e_i\|$
5: **end for**
6: **for** each $l = 1, \dots, N$ **do**
7: $\quad a_l \leftarrow \sum \big\{ \eta_N(y_j) \ \big| \ \mathcal{I}(j) = l \big\}$
8: $\quad b_l \leftarrow \#\big\{ j \ \big| \ \mathcal{I}(j) = l \big\}$
9: **end for**
10: $\tilde{\xi}_N \leftarrow \sum_{l=1}^{N} \frac{a_l}{b_l} \mathbf{1}_{A_l}$

4. Examples

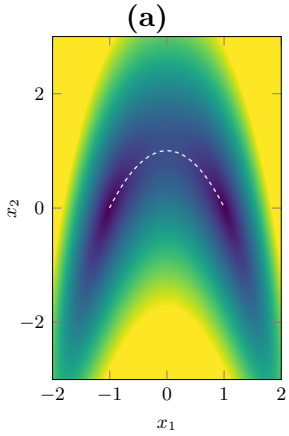
4.1. Curved double well potential

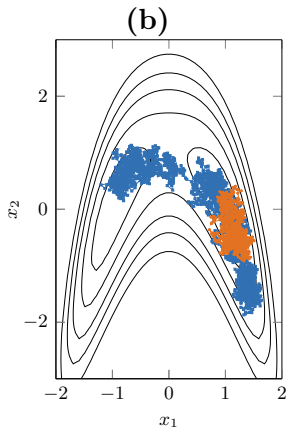
As our first example, we consider a diffusion process in a two-dimensional double well potential with curved reaction pathway that was first studied in the context of reaction coordinates in [18]. The system obeys the SDE

$$\mathrm{d}\mathbf{X}_t = -\nabla V(\mathbf{X}_t)\mathrm{d}t + \sqrt{2\beta^{-1}}\mathrm{d}\mathbf{W}_t \ , \tag{14}$$

where V is called the potential, β is the non-dimensionalized inverse temperature, and $\mathbf{W}_t$ is a standard Wiener process. The potential with its two wells can be seen in Figure 2 (a). The wells around the minima $(\pm 1, 0)$ are connected by the minimum energy path $\{x_2 = 1 - x_1^2, \ x_1 \in [-1, 1]\}$. Using Algorithm 3.2, we aim to compute the

Voronoi-Galerkin approximation of the reaction coordinate $\tilde{\xi}_N$ that hopefully resolves the transition pathway and retains the dominant implied timescales of the system. The algorithm was implemented in Matlab, and for the time-critical parts highly-optimized native routines (such as `dsearchn` for the nearest-neighbor search) were used.





(c)

t_0	∞
t_1	5.9332
t_2	0.9021
t_3	0.6031
t_4	0.4001

Figure 2: (a) Curved double well potential with minimum energy path. (b) Trajectory of length 2 (red) and 6 (blue). (c) Implied timescales of the full process.

Setup. The implied timescales[5] of the full process for $\beta = 0.5$ can be seen in Figure 2 (c). As expected, the system is timescale-separated, with the single slow timescale representing the mean expected waiting time for a single transition between the wells [3]. We also see that the relaxation time $t = 2$ falls in between the slow and fast timescales, so we use it as input for Algorithm 3.2. Indeed, typical trajectories of length 2.0 seem to sample the local metastable set, yet undergo no transition (Figure 2 (b)). Moreover, we assume the dimension $r = 1$ of the effective dynamics to be known.

The transition matrix was computed based on dynamical data that was extracted from a single well-equilibrated trajectory $\mathbb{T} = \{x_0, \Phi^\tau x_0, \ldots\}$, with starting point $x_0 = (-1, 0)$, step size $\tau = 10^{-2}$ and $2 \cdot 10^7$ steps. We constructed data sets $\mathbb{X}_M$, $\mathbb{Y}_M$ of form (8) with $M \approx 2 \cdot 10^7$.

The subset of the rectangle $[-2, 2] \times [-3, 3]$ that is covered by $\mathbb{X}_M$ will be taken as the "interesting" state space region $\hat{\mathbb{X}}$ where we want to compute $\tilde{\xi}_N$. It is partitioned into $N = 1000$ Voronoi cells A_k with centers computed by both the k-means algorithm (provided by Matlabs `kmeans` function) as well as Algorithm 3.1 (see Figure 3 (a)). While the center points based on Algorithm 3.1 cover $\hat{\mathbb{X}}$ evenly (by construction), the k-means-based center points appear to emphasize the metastable regions and slightly under-sample the transition regions. As described in Section 3.2, indicator functions over the A_k will serve as the basis for the Galerkin discretization.

[5] Computed by a fine Ulam approximation of the eigenvalues of $\mathcal{T}^t$ and using the relation (1). Numerical errors resulting from finite dynamical data introduce a dependence of the timescales on the lag time t in (1). We compute the timescales at $t = 2$, as here they are locally almost constant in t.

The $2r + 1 = 3$ generic observables $\eta_i : \mathbb{R}^2 \to \mathbb{R}$ were chosen as linear functions $\eta_i(x) = a_i^\intercal x$ with randomly-chosen orthogonal coefficients $a_i \in \mathbb{R}^2$. They were generated in Matlab by applying Gram Schmidt orthogonalization to the rows of a randomly drawn matrix:

```
rng(1); A = rand(3,2); A = GramSchmidt(A) .
```

However, as in this toy example the embedding dimension is higher than the original dimension, the third coefficient could not be chosen orthogonal to the other two. Instead, $a_3 = (1, 0)^\intercal$ was chosen for simplicity.

Results and analysis. Figure 3 (a) shows the computed reaction coordinate $\tilde{\xi}_N$, evaluated at the center points of the Voronoi cells. Note that $\tilde{\xi}_N$ is defined and can be evaluated on all of $\mathbb{X}$, but is constant on the individual Voronoi cells, thus $\tilde{\xi}_N(\mathbb{X})$ is finite.

The points appear to be concentrated around a one-dimensional manifold, which is the transition manifold $\mathbb{M}$ embedded into $\mathbb{R}^3$ by η. As detailed in [4], a standard manifold learning method, such as the well-known diffusion maps algorithm [9, 29, 30], can be used to parametrize the embedded points along this manifold (see the color-coding in Figure 3 (b)). Calling the parametrization by the leading diffusion maps eigenvector $\Psi_1 : \tilde{\xi}_N(\mathbb{X}) \to \mathbb{R}$, the map

$$\hat{\xi}_N(x) := \Psi_1\big(\tilde{\xi}_N(x)\big)$$

can then be used as one-dimensional reaction coordinate. Figure 3 (b) shows that for both types of Voronoi center points, $\hat{\xi}_N$ parametrizes the transition pathway.

To verify the quality of the one-dimensional reaction cooridinate $\hat{\xi}_N$, we compare the timescales of the full process X_t and the projected process $\hat{\xi}_N(X_t)$. The latter is computed by using the projected trajectory $\hat{\xi}_N(\mathbb{T})$ as dynamical data for a Ulam approximation of the projected transfer operator $\mathcal{T}^t_{\hat{\xi}_N}$ (on a regular 50 interval partition of $[\min \hat{\xi}_N(\mathbb{T}), \max \hat{\xi}_N(\mathbb{T})]$) and using relation (1). Table 1 shows that the reaction coordinates based on the Voronoi centers of both k-means and Algorithm 3.1 significantly outperform the naively-chosen reaction coordinate $\zeta(x_1, x_2) = x_1$. Moreover, despite the seemingly worse approximation behaviour at the borders of $\hat{\mathbb{X}}$, the reaction coordinate based on Algorithm 3.1 seems to preserve the dominant timescales better. A possible reason might be the better resolution in the transition region.

	k-means		**picking algorithm**		**x_1-coordinate**	
	timescale	rel. error	timescale	rel. error	timescale	rel. error
t_1	5.8899	0.0073	5.9034	0.0050	5.7130	0.0371
t_2	0.8615	0.0450	0.8789	0.0258	0.7964	0.1172
t_3	0.5625	0.0673	0.5838	0.0320	0.5380	0.1079
t_4	0.3746	0.0638	0.3906	0.0238	0.3444	0.1394

Table 1: Implied timescales of the curved double well potential under projection onto different reaction coordinates.

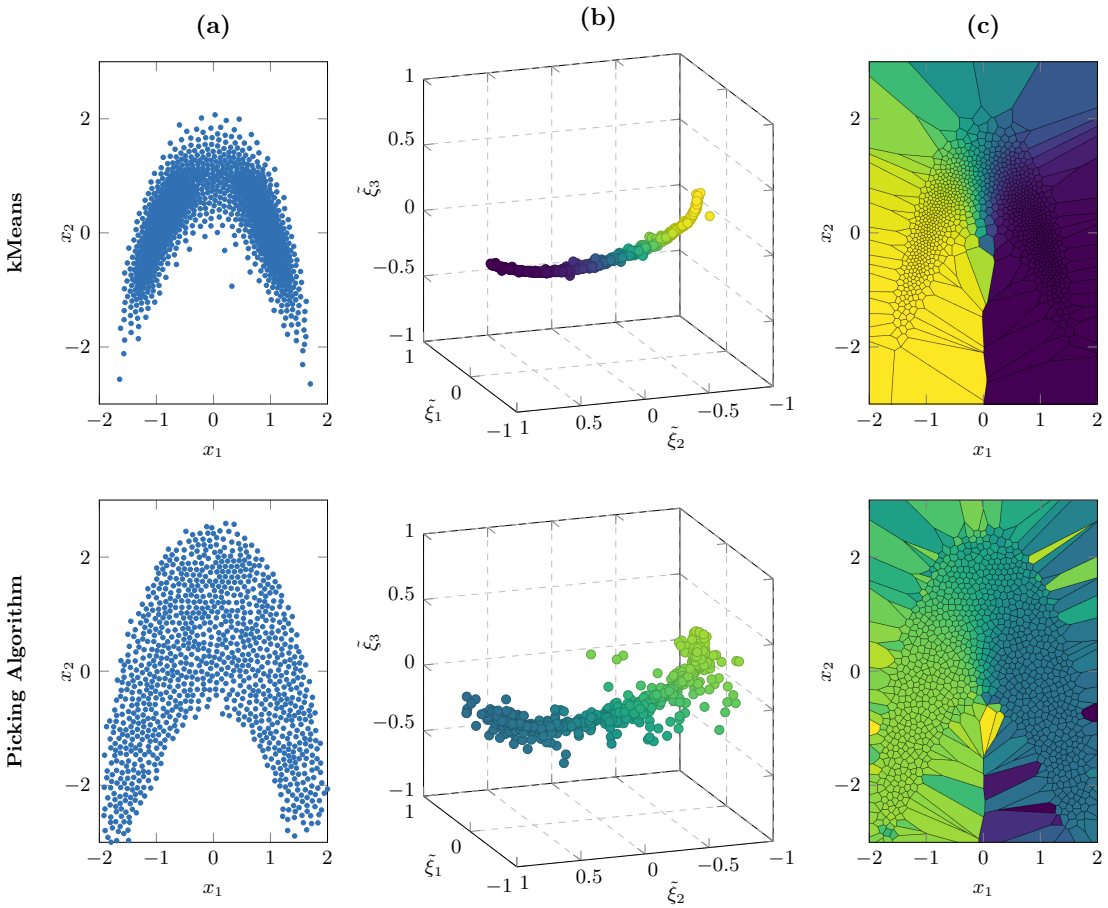


Figure 3: (a) Voronoi center points, computed with k-means clustering and the picking algorithm, respectively. (b) The three components of the computed reaction coordinate $\tilde{\xi}_N$, evaluated at the respective Voronoi centers. The coloring indicates the dominant diffusion maps eigenvector, which is used to construct the one-dimensional reaction coordinate $\hat{\xi}_N$. (c) $\hat{\xi}_N$ as a continuous function on $\mathbb{X}_M$.

4.2. Dialanine

Finally, we show that Algorithm 3.2 can be used to sucessfully identify good reaction coordinates in realistic molecular systems. At temperature 400K, the peptide Dialanine in aqueous solution shows four metastable conformations that can be described by two essential backbone dihedral angles φ, ψ (Figure 4). We examine whether our algorithm can identify reaction coordinates that correlate with φ, ψ and reproduce the dominant timescales.

Setup. The molecule consists of 22 atoms (including hydrogen), thus the state space $\mathbb{X}$ is 66-dimensional. The relaxation time $t = 20$ ps as well as the embedding dimension

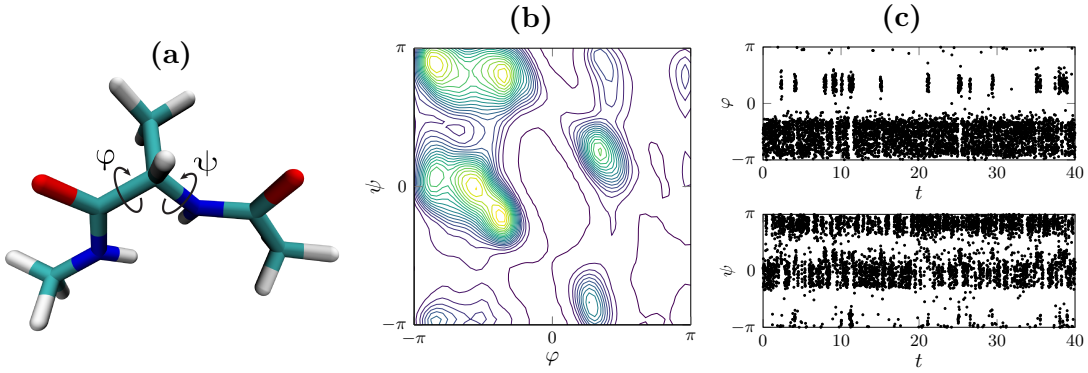


Figure 4: (a) Dialanine with its two essential dihedral angles φ and ψ. (b) Ramachandran plot of φ, ψ, revealing four metastable conformations. (c) Frames from a long trajectory projected onto φ and ψ.

$r = 2$ are assumed to be known. For the dynamical data, a long molecular trajectory $\mathbb{T}$ of Dialanine in explicit water was computed using the MD software Gromacs. To ensure the internal conformational switching to be the dominant motion of the system, the global translational and rotational degrees of freedom (that describe the molecule "floating around" in the simulation box) were removed from the trajectory prior to further analysis. The simulation of overall length 40 ns (step size 0.002 ps) yielded data sets $\mathbb{X}_M$ and $\mathbb{Y}_M$ of size $M \approx 2 \cdot 10^6$.

Again, $N = 1000$ Voronoi centers in the region covered by $\mathbb{X}_M$ were computed using the k-means algorithm and Algorithm 3.1. The projection of these points onto the (φ, ψ)-plane can be seen in Figure 5 (b). Again, the former emphasizes the metastable sets, whereas the latter covers the range of values more evenly.

For the embedding functions $\eta : \mathbb{R}^{66} \to \mathbb{R}^5$, linear functions with orthogonal coefficient vectors were chosen as in Section 4.1.

Results and analysis. Figure 5 visualizes the computed reaction coordinates with Voronoi center points choosen by the k-means algorithm (top) and Algorithm 3.1 (bottom). The first three of the five components of $\tilde{\xi}_N$, evaluated at the Voronoi center points, are plotted in Figure 5 (a). The two dominant diffusion map eigenvectors on the data points, Ψ_1, Ψ_2, are color-coded, and have again been used to construct the two-dimensional reaction coordinate

$$\hat{\xi}_N(x) := \begin{pmatrix} \Psi_1\big(\tilde{\xi}_N(x)\big) \\ \Psi_2\big(\tilde{\xi}_N(x)\big) \end{pmatrix} .$$

As visualized in Figure 5 (b), $\hat{\xi}_N$ shows a clear correlation with the dihedral angles ϕ, ψ for both methods of choosing the Voronoi center points. The range of $\hat{\xi}_N$ can be seen in Figure 5 (c).

We again compute the implied timescales of the reduced process $\hat{\xi}_N(X_t)$. To yield higher accuracy than the simple box-based Ulam method from the previous example,

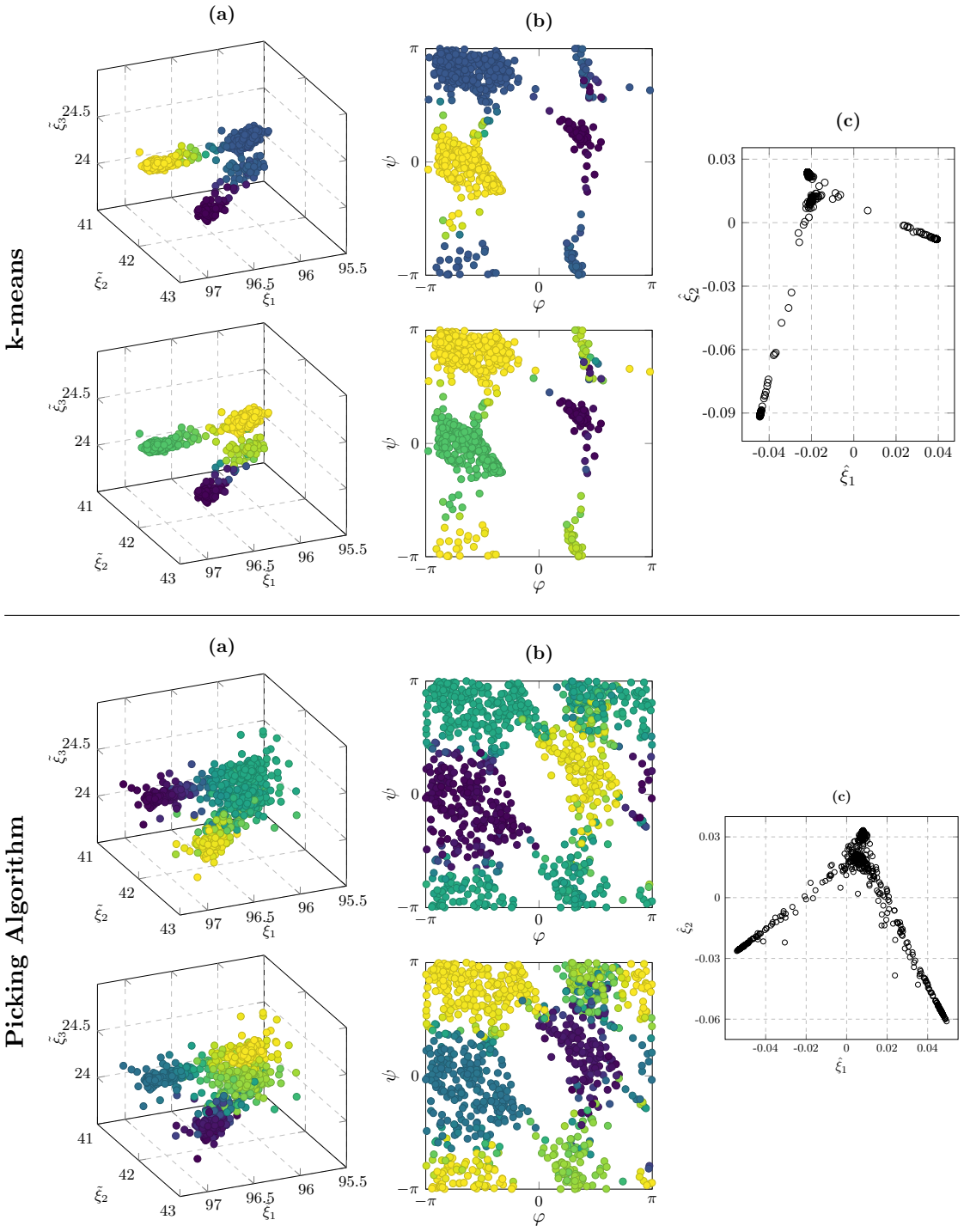


Figure 5: Dialanine reaction coordinates. (a) Embedded transition manifold. Colored are the diffusion map eigenvectors Ψ_1, Ψ_2 that are used to construct the two-dimensional reaction coordinate $\hat{\xi}_N$ (b) Corellation between $\hat{\xi}_N$ and the two dihedral angles. (c) $\hat{\xi}_N$ evaluated on all Voronoi centers, i.e. all possible values of $\hat{\xi}_N$. The four clusters correspond to the four metastable conformations from Figure 4 (b).

we utilize the PyEMMA software package [25] with its built-in methods to discretise the transfer operator, estimate its eigenvalues and compute the timescales. Again the implied timescales show a certain dependence on the lag time parameter t (see Equation (1)) due to numerical inaccuracies. We compute the timescales at the lag time of minimal local variance, $t = 31.96$ps.

Computing the timescales of the full 66-dimensional process with the necessary accuracy is numerically infeasible, so a rigorous error analysis is not possible for this system. Instead, we utilize the variational principle of conformation dynamics [23] which states that the timescales of the full process are always *underestimated* by those of any projection of the process. Thus, larger projected timescales can in general be considered more accurate. However, due to the possibility of systematic errors in approximating the projected timescales (discretisation of the transfer operator, finite amount of dynamical data), this variational principle might be violated in unvetted ways. Thus, we additionally offer a comparison to the timescales of a manually-chosen RC that can be assumed to be "good", namely the backbone dihedrals φ, ψ. We emphasise that these again only represent an approximation to the full system's timescales with unknown error.

Using these two error estimators, we compare our RCs $\hat{\xi}_N$ for both the k-means and the picking algorithm to a two-dimensional TICA projection, a dimensionality reduction method that is popular in MD analysis [24]. The method finds the directions in the data sets with maximal global autocorrelation for a specified lag time, and thus always constructs *linear* reaction coordinates. For this lag time $\tau = 120$ps was chosen as it maximizes the cumulative kinetic variance (95.5%) [22].

The three (nontrivial) dominant timescales and their deviation from the benchmark (φ, ψ)-projection can be seen in Table 2. The remaining timescales $t_i,\ i \geq 4$ are significantly smaller (< 5ps) and are thus considered irrelevant.

Judging by both the variational principle and the deviation from the benchmark projection, the k-means Galerkin-RC provides a better approximation of the first two dominant timescales than the TICA projection. The picking algorithm Galerkin-RC yields a worse approximation of the first, but a better aproximation of the second timescale, compared to TICA.

However, both the new k-means and the picking algorithm Galerkin-RCs significantly outperform the TICA projection in approximating the third dominant timescale. The process $\hat{\xi}_N(X_t)$ can thus be considered a significantly more realistic representation of the slow processes in X_t than the TICA-projected process.

5. Conclusion

In this paper we introduced a Galerkin approach to the computation of the ε-optimal reaction coordinates introduced in [4]. The Galerkin approach is suitable whenever the transition matrix of a Markov State Model can be computed from available simulation data, e.g. if the data consists of a long equilibrated trajectory. As with any Galerkin method, one has to choose a set of basis functions, and this choice is crucial in order to obtain a small approximation error. We provide two algorithms in order to compute

(a) dominant timescales

timescale	t_1	t_2	t_3
k-means	193.10	62.38	41.63
picking	187.61	61.73	41.25
TICA	191.78	61.27	29.84
(φ, ψ)	194.71	62.93	41.27

(b) rel. error to (φ, ψ)-projection

timescale	t_1	t_2	t_3
k-means	0.0083	0.0088	0.0088
picking	0.0365	0.0192	0.0004
TICA	0.0150	0.0264	0.2769

Table 2: (a) Implied timescales of the Dialanine system under projection onto different reaction coordinates. (b) Relative error to the (φ, ψ)-projection.

a basis of Voronoi ansatz functions directly from the data. Both algorithms are highly scalable and readily available, making the Voronoi based Galerkin method straightforward to apply for practicioners who have existing simulation data on their hard drives. We showed that the resulting approximation error in the dominant time scales is competitive with state of the art dimension reduction techniques. This demonstrates that the reaction coordinates we compute here can be used to build efficient coarse grained models. Of course the computed reaction coordinates themselves are also of independent value. In the case of the Dipeptide Alanine we showed that our computed reaction coordinates and the dihedral angles, which are typically used as reaction coordinates for this system, produce a similar portrait of the system when viewed in this reduced space.

The requirement to have simulation data that samples the invariant distribution μ is of course somewhat strict, and in future work we will relax this requirement to a "local" version, i.e. we will work with samples that are locally equilibrated in some metastable region of the state space.

Acknowledgements

This research has been funded by Deutsche Forschungsgemeinschaft (DFG) through grant CRC 1114 "Scaling Cascades in Complex Systems", Project B03 "Multilevel coarse graining of multi-scale problems".

References

[1] D. Aristoff and T. Lelievre. Mathematical analysis of temperature accelerated dynamics. 2014.

[2] J. R. Baxter and J. S. Rosenthal. Rates of convergence for everywhere-positive Markov chains. *Statistics & probability letters*, 22(4):333–338, 1995.

[3] A. Bianchi, A. Bovier, and D. Ioffe. Pointwise estimates and exponential laws in metastable systems via coupling methods. *The Annals of Probability*, 40(1):339–371, 01 2012.

[4] A. Bittracher, P. Koltai, S. Klus, R. Banisch, M. Dellnitz, and C. Schütte. Transition manifolds of complex metastable systems: Theory and data-driven computation of effective dynamics. *arXiv preprint arXiv:1704.08927*, 2017.

[5] G. R. Bowman, V. S. Pande, and F. Noé, editors. *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, volume 797 of *Advances in Experimental Medicine and Biology*. Springer, 2014.

[6] R. Bridson. Fast poisson disk sampling in arbitrary dimensions. In *ACM SIGGRAPH 2007 Sketches*, SIGGRAPH '07, New York, NY, USA, 2007. ACM.

[7] C. J. Camacho and D. Thirumalai. Kinetics and thermodynamics of folding in model proteins. *Proceedings of the National Academy of Sciences*, 90(13):6369–6372, 1993.

[8] J. D. Chodera and F. Noé. Markov state models of biomolecular conformational dynamics. *Current Opinion in Structural Biology*, 25:135 – 144, 2014.

[9] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1):5–30, 2006.

[10] P. Deuflhard, W. Huisinga, A. Fischer, and C. Schütte. Identification of almost invariant aggregates in reversible nearly uncoupled markov chains. *Linear Algebra and its Applications*, 315(13):39 – 59, 2000.

[11] A. K. Faradjian and R. Elber. Computing time scales from reaction coordinates by milestoning. *J. Chem. Phys.*, 120:10880–10889, 2004.

[12] G. Froyland, G. Gottwald, and A. Hammerlindl. A computational method to extract macroscopic variables and their dynamics in multiscale systems. *SIAM Journal on Applied Dynamical Systems*, 13(4):1816–1846, 2014.

[13] W. R. Gilks, S. Richardson, and D. Spiegelhalter. *Markov chain Monte Carlo in practice*. CRC press, 1995.

[14] B. Hunt and V. Kaloshin. Regularity of embeddings of infinite-dimensional fractal sets into finite-dimensional spaces. *Nonlinearity*, 12(5):1263—-1275, 1999.

[15] O. Junge and P. Koltai. Discretization of the Frobenius–Perron operator using a sparse Haar tensor basis: the sparse Ulam method. *SIAM Journal on Numerical Analysis*, 47(5):3464–3485, 2009.

[16] B. O. Koopman. Hamiltonian systems and transformation in hilbert space. *Proceedings of the National Academy of Sciences*, 17(5):315–318, 1931.

[17] A. Laio and F. L. Gervasio. Metadynamics: a method to simulate rare events and reconstruct the free energy in biophysics, chemistry and material science. *Rep. Prog. Phys.*, 71(12):126601, 2008.

[18] F. Legoll and T. Lelièvre. Effective dynamics using conditional expectations. *Nonlinearity*, 23(9):2131, 2010.

[19] W. Li and M. A. Recent developments in methods for identifying reaction coordinates. *Molecular simulation*, 40(10-11), 2014.

[20] J. Lu and E. Vanden-Eijnden. Exact dynamical coarse-graining without time-scale separation. *The Journal of chemical physics*, 141(4):07B619_1, 2014.

[21] I. Mezić. Spectral properties of dynamical systems, model reduction and decompositions. *Nonlinear Dynamics*, 41(1):309–325, 2005.

[22] F. Noe and C. Clementi. Kinetic distance and kinetic maps from molecular dynamics simulation. *J. Chem. Theory Comput.*, 11(10):5002–5011, 2015.

[23] F. Noé and F. Nüske. A variational approach to modeling slow processes in stochastic dynamical systems. *Multiscale Model. Simul.*, 11(2):635–655, 2013.

[24] G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis, and F. Noé. Identification of slow molecular order parameters for Markov model construction. *J. Chem. Phys.*, 139(1):015102, 2013.

[25] M. K. Scherer, B. Trendelkamp-Schroer, F. Paul, G. Pérez-Hernández, M. Hoffmann, N. Plattner, C. Wehmeyer, J.-H. Prinz, and F. Noé. Pyemma 2: A software package for estimation, validation, and analysis of markov models. *Journal of Chemical Theory and Computation*, 11(11):5525–5542, 2015. PMID: 26574340.

[26] C. Schütte and M. Sarich. *Metastability and Markov State Models in Molecular Dynamics.* Courant Lecture Notes in Mathematics, 2013.

[27] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard. A direct approach to conformational dynamics based on hybrid monte carlo. *Journal of Computational Physics*, 151(1):146 – 168, 1999.

[28] D. E. Shaw, M. M. Deneroff, R. O. Dror, J. S. Kuskin, R. H. Larson, J. K. Salmon, C. Young, B. Batson, K. J. Bowers, J. C. Chao, M. P. Eastwood, J. Gagliardo, J. P. Grossman, C. R. Ho, D. J. Ierardi, I. Kolossváry, J. L. Klepeis, T. Layman, C. McLeavey, M. A. Moraes, R. Mueller, E. C. Priest, Y. Shan, J. Spengler, M. Theobald, B. Towles, and S. C. Wang. Anton, a special-purpose machine for molecular dynamics simulation. *Commun. ACM*, 51(7):91–97, July 2008.

[29] A. Singer and R. R. Coifman. Non-linear independent component analysis with diffusion maps. *Applied and Computational Harmonic Analysis*, 25(2):226–239, 2008.

[30] A. Singer, R. Erban, I. G. Kevrekidis, and R. R. Coifman. Detecting intrinsic slow variables in stochastic dynamical systems by anisotropic diffusion maps. *Proceedings of the National Academy of Sciences*, 106(38):16090–16095, 2009.

[31] N. Socci, J. N. Onuchic, and P. G. Wolynes. Diffusive dynamics of the reaction coordinate for protein folding funnels. *The Journal of chemical physics*, 104(15):5860–5868, 1996.

[32] E. Vanden-Eijnden. Transition path theory. *Computer Simulations in Condensed Matter Systems: From Materials to Chemical Biology Volume 1*, pages 453–493, 2006.

[33] M. Weber. *Meshless Methods in Conformation Dynamics*. PhD thesis, FU Berlin, 2006.

[34] M. Weber, K. Fackeldey, and C. Schütte. Set-free Markov state model building. *J. Chem. Phys.*, 146:124133, 2017.

A. Burst-based algorithm

We state an algorithm proposed in [4] for the pointwise computation of $\tilde{\xi}$ at the evaluation points $\{x_1, \ldots, x_N\}$. The algorithm requires M short trajectories starting at each of the N evaluation points x_i (NM trajectories in total).

Algorithm 1.1 Point-wise computation of the TMRC

Input: evaluation points $\{x_1, \ldots, x_N\}$, TM dimension r, lag time t
Output: $\{\tilde{\xi}(x_1), \ldots, \tilde{\xi}(x_N)\}$

1: Choose a generic observable $\eta : \mathbb{X} \to \mathbb{R}^{2r+1}$ with coefficients $\eta_i \in L^\infty(\mathbb{X})$.
2: **for** each evaluation point x_i **do**
3: **for** $k = 1, \ldots, M$ **do**
4: compute a realization of the stochastic dynamics of length t:
 $y_i^{(k)} \leftarrow \Phi^t x_i$
5: **end for**
6: Compute the approximation to (5) by a Monte Carlo sum:
 $$\tilde{\xi}(x_i) \leftarrow \frac{1}{M} \sum_{k=1}^{M} \eta\big(y_i^{(k)}\big)$$
7: **end for**

B. Objective functions

We show that the objective functions discussed in section 3.2 minimize lower bounds of the L^2 error and the uniform error of $\tilde{\xi}$ respectively.

k-means objective function. We show that the objective function $S_\xi(A_1, \ldots, A_N)$ defined in (12) is an unbiased estimator of $\|\tilde{\xi} - \tilde{\xi}_N\|^2_{L^2_\mu(\hat{\mathbb{X}})}$. First, since $x_i \sim \mu$,

$$\mathbf{E}\left[\bar{\xi}_{A_k}\right] = \mathbf{E}\left[\frac{1}{|A_k|}\sum_{x_i \in A_k} \tilde{\xi}(x_i)\right] = \frac{\langle \mathbf{1}_{A_k}, \tilde{\xi}\rangle_\mu}{\langle \mathbf{1}_{A_k}, \mathbf{1}\rangle_\mu}.$$

Thus

$$\begin{aligned}\mathbf{E}\left[\sum_{x_i \in A_k} \|\tilde{\xi}(x_i) - \bar{\xi}_{A_k}\|^2_{\mathbb{R}^{2k+1}}\right] &= \int_{A_k}\left(\tilde{\xi}(x) - \frac{\langle \mathbf{1}_{A_k}, \tilde{\xi}\rangle_\mu}{\langle \mathbf{1}_{A_k}, \mathbf{1}\rangle_\mu}\right)^2 d\mu(x) \\ &= \int_{A_k}\left(\tilde{\xi}(x) - \tilde{\xi}_N(x)\right)^2 d\mu(x)\end{aligned}$$

where the last line follows from $\tilde{\xi}_N = \sum_k \frac{\langle \mathbf{1}_{A_k}, \tilde{\xi}\rangle_\mu}{\langle \mathbf{1}_{A_k}, \mathbf{1}\rangle_\mu}\mathbf{1}_{A_k}$. Summing over k then gives

$$\mathbf{E}S_\xi(A_1, \ldots, A_N) = \|\tilde{\xi} - \tilde{\xi}_N\|^2_{L^2_\mu(\hat{\mathbb{X}})},$$

as desired.

Uniform error objective function. Evidently, we have

$$\|\tilde{\xi} - \tilde{\xi}_N\|_{L^\infty(A_k)} = \sup_{x \in A_k} \|\tilde{\xi}(x) - \bar{\xi}_{A_k}\|$$

with $\bar{\xi}_{A_k}$ as above. Let now $e_k \in A_k$ be such that $\tilde{\xi}(e_k) = \bar{\xi}_{A_k}$ (such an e_k exists by continuity of $\tilde{\xi}$). Then

$$\|\tilde{\xi} - \tilde{\xi}_N\|_{L^\infty(A_k)} = \sup_{x \in A_k} \|\tilde{\xi}(x) - \tilde{\xi}(e_k)\| \leq L \sup_{x \in A_k} \|x - e_k\| \leq L\,\mathrm{diam}(A_k).$$

Since $A_1, \ldots, A_N$ partition $\hat{\mathbb{X}}$, we have

$$\|\tilde{\xi} - \tilde{\xi}_N\|_{L^\infty(\hat{\mathbb{X}})} = \max_{k=1\ldots N} \|\tilde{\xi} - \tilde{\xi}_N\|_{L^\infty(A_k)} \leq L \max_{k=1\ldots N} \mathrm{diam}(A_k).$$