Galerkin Approximation of Dynamical Quantities using Trajectory Data

Aaron Dinner and Erik Thiede
University of Chicago

Dimitrios Giannakis and Jonathan Weare
New York University

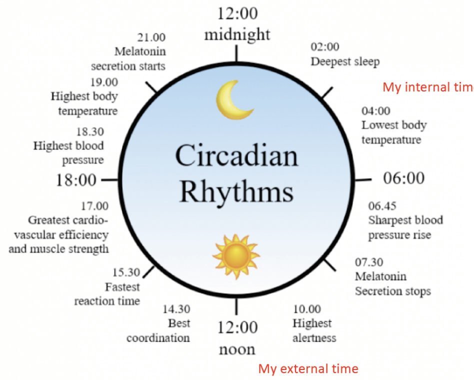
Funding from NIH and NSF MolSSI

December 5, 2018

Motivation

My group and I are interested in understanding how molecular interactions give rise to the complex behavior of living systems.

For example, we want to understand how circadian clocks precisely time daily physiological rhythms.



My internal time

My external time

Article

Current Biology

Geographically Resolved Rhythms in Twitter Use Reveal Social Pressures on Daily Activity Patterns

Highlights

- Temporal and geographical patterns of Twitter use reveal widespread social jet lag
- Magnitude of social jet lag is correlated with both geography and lifestyle factors

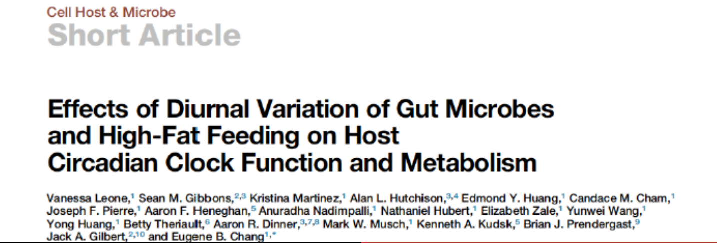
Authors

Eugene Leypunskiy, Emre Kıcıman, Mili Shah, Olivia J. Walch, Andrey Rzhetsky, Aaron R. Dinner, Michael J. Rust

JBR PERSPECTIVES ON DATA ANALYSIS

Bootstrapping and Empirical Bayes Methods Improve Rhythm Detection in Sparsely Sampled Data

Alan L. Hutchison,[*,†,‡,1] Ravi Allada,[§] and Aaron R. Dinner[‡,||,¶,1]



Cell Host & Microbe

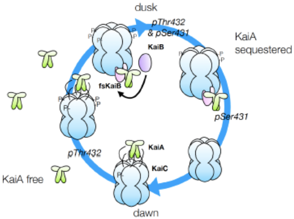
Short Article

Effects of Diurnal Variation of Gut Microbes and High-Fat Feeding on Host Circadian Clock Function and Metabolism

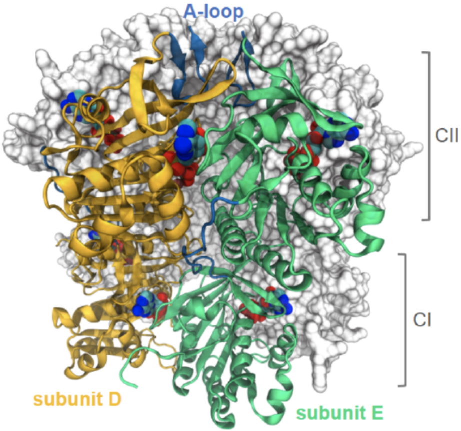
Vanessa Leone,[1] Sean M. Gibbons,[2,3] Kristina Martinez,[1] Alan L. Hutchison,[3,4] Edmond Y. Huang,[1] Candace M. Cham,[1] Joseph F. Pierre,[1] Aaron F. Heneghan,[5] Anuradha Nadimpalli,[1] Nathaniel Hubert,[1] Elizabeth Zale,[1] Yunwei Wang,[1] Yong Huang,[1] Betty Theriault,[6] Aaron R. Dinner,[3,7,8] Mark W. Musch,[1] Kenneth A. Kudsk,[5] Brian J. Prendergast,[9] Jack A. Gilbert,[2,10] and Eugene B. Chang[1,*]

Motivation: A molecular problem

We are using molecular dynamics simulations to study the core oscillator of the simplest known circadian clock, the Kai system from cyanobacteria.



- How does KaiA promote phosphorylation (via nucleotide exchange)?
- How is the ordered sequence of phosphorylation states generated?
- How does phosphorylation in CII lead to KaiB binding to CI?



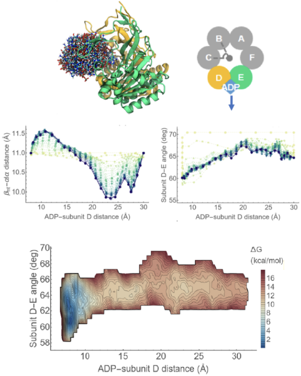
Motivation: Challenges for molecular dynamics

Using a pipeline of enhanced sampling methods, we can obtain pathways for nucleotide exchange from the ATPase KaiC, and a free energy surface.

- Locally enhanced sampling to identify direction of release.
- Steered molecular dynamics to generate an initial pathway.
- String calculations in a space of 9 collective variables.
- Umbrella sampling in 2 collective variables.

With (132 Å)^3 box of water and KCl, ≈217,000 atoms.

Simulations by Lu Hong



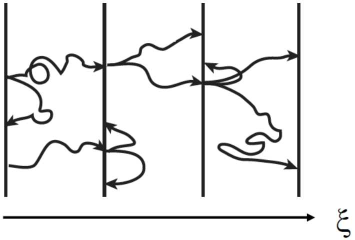
Hong, Vani, Thiede, Rust, Dinner, PNAS, Latest Articles (Nov 15)

However, we would like to obtain kinetic quantities in addition to thermodynamic ones.

Trajectory splitting for sampling

One general strategy for obtaining dynamical statistics is to branch and prune trajectories based on collective variables that describe the dynamics of interest.

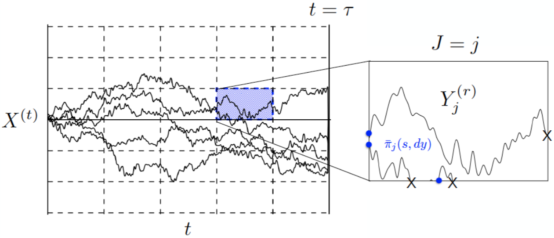
The key idea is that sampling the statistics of the regions is simpler than sampling those of the overall process.



There are many such splitting methods (e.g., FFS, Weighted Ensemble, STePS, NEUS, etc.).

Trajectory splitting and stratification for sampling

We have developed a splitting scheme that performs stratified sampling for trajectories (nonequilibrium umbrella sampling; Warmflash *et al.*, 2007; Dickson *et al.*, 2009-2011).



See for mathematical discussion and improved algorithm Dinner, Mattingly, Tempkin, Van Koten, Weare, SIAM Rev 60, 909-938 (2018).

Markov State Models (MSMs)

MSMs are a popular alternative. In this scheme, one stitches together information from existing trajectories by constructing a transition matrix.

$$\bar{P}_{ij} = \frac{\sum_{n=1}^{N} \mathbf{1}_{S_j}(Y_n)\,\mathbf{1}_{S_i}(X_n)}{\sum_{n=1}^{N} \mathbf{1}_{S_i}(X_n)}$$

where **1** is the indicator function

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

and $\mu(x)$ is the probability distribution of initial states.

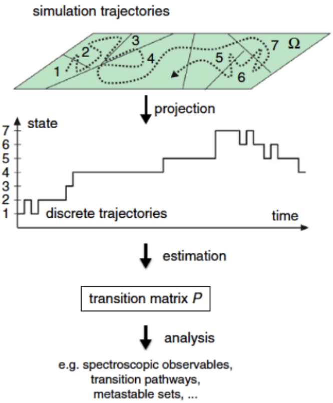


Figure from Chodera and Noé (2014)
Current Opinion in Structural Biology 25, 135.

Dynamical operators

Rather than consider the evolution of points in space, we can instead consider the evolution of functions of those points.

The transition operator with lag time of *s* is defined as

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

where *f* is a function on the state space.

We can then write the MSM transition matrix as

$$P_{ij} = \frac{\int \mathbf{1}_{S_i}(x) \mathcal{K}_s \mathbf{1}_{S_j}(x) \mu(dx)}{\int \mathbf{1}_{S_i}(x) \mu(dx)}.$$

The goal of this talk is to develop a theory for obtaining dynamical statistics from equations of such an operator.

The generator

In practice, we work with the generator:

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

The generator shows how expectations of functions change over time. It is a (discrete-time) stochastic analog of the Liouvillian.

Let's consider some examples to see how dynamical statistics can be defined in terms of this operator.

Mean first passage time (MFPT)

The MFPT is the expectation of τ_A, conditioned on the dynamics starting at x:

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

where $\Xi^{(t)}$ is a time-homogeneous Markov process.

The MFPT obeys the operator equation

$$\mathcal{L} m_A(x) = -1 \text{ for } x \text{ in } A^c$$
$$m_A(x) = 0 \text{ for } x \text{ in } A.$$

Here A^c denotes the set of all states not in A.

Mean first passage time (MFPT)

To see this, note that for all x in A^c we have

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

Rearranging gives

$$\mathcal{L} m_A(x) = \frac{\mathcal{K}_{\Delta t} m_A(x) - m_A(x)}{\Delta t} = -1$$

The committor (q_+)

The forward committor is defined as the probability of entering B before A, conditioned on starting at x:

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

We can show that the forward committor obeys

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

by similar arguments.

The committor (q_+)

We introduce the random variable

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

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

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

which gives $\mathcal{L}q_+(x) = 0$ on rearranging.

General form

The examples above follow the general form

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

Similar equations connecting short- and long-time behavior exist for reactive fluxes, autocorrelation times, etc.

Given an equation for a desired quantity, we want to solve it numerically.

Overview of numerical solution

We construct an approximation of the operator equation through the following steps.

1. *Homogenize the boundary conditions:* Use a guess for g to rewrite the problem with homogeneous boundary conditions ($b = 0$).
2. *Construct a Galerkin scheme:* Approximate the solution as a sum of basis functions and convert the result of step 1 into a matrix equation.
3. *Approximate inner products with trajectory averages:* Estimate needed terms from trajectory averages and solve for g.

Homogenizing the boundary conditions

To make the space closed to linear combinations of functions satisfying the boundary conditions, we introduce a guess function r that is equal to b on D^c and replace the quantity of interest (g) with the difference from the guess:

$$\gamma(x) = g(x) - r(x)$$

and

$$\begin{aligned}\mathcal{L}\gamma(x) &= h(x) - \mathcal{L}r(x) \text{ for } x \text{ in } D\\ \gamma(x) &= 0 \text{ for } x \text{ in } D^c.\end{aligned}$$

A naive guess can always be constructed as

$$r(x) = \mathbf{1}_{D^c}(x)b(x),$$

but if possible, r should be chosen so that γ can be efficiently expressed using the basis functions.

Constructing a Galerkin scheme

Expand γ in terms of basis functions that satisfy the homogenized boundary conditons.

$$\gamma(x) \approx \sum_{j=1}^{M} a_j \phi_j(x).$$

Substitute, multiply by $\phi_i(x)$ and an *arbitrary* initial density $\mu(x)$ and integrate

$$\sum_{j=1}^{M} a_j \int \phi_i(x)\phi_j(x)\mu(x)dx = \int \phi_i(x)h(x)\mu(x)dx - \int \phi_i(x)r(x)\mu(x)dx$$

Constructing a Galerkin scheme

Or, more compactly,

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

where

$$L_{ij} = \left\langle \phi_i, \mathcal{L}\phi_j \right\rangle, h_i = \left\langle \phi_i, h \right\rangle, \text{and } r_i = \left\langle \phi_i, r \right\rangle.$$

and

$$\langle u, v \rangle = \int u(x)v(x)\mu(dx).$$

Approximating inner products with trajectory averages

In practice, cover the space with initial points and run (short) simulations. Much of the power of the method comes from the fact that the initial distribution can be arbitrarily chosen.

Draw pairs of points (X_n, Y_n) separated by time interval Δt and compute

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

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

Pseudocode

1. Sample N pairs of configurations (X_n, Y_n) where Y_n is the configuration resulting from propagating the system forward from X_n for time Δt.
2. Construct a set of M basis functions ϕ_i and, if needed, the guess function r.
3. Estimate the matrix elements L_{ij}, h_i, and r_i from the data.
4. Solve the resulting matrix equations for the coefficients a_j and construct an estimate of the function of interest:

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

Relation to MSMs

The procedure is closely related to other approaches (EDMD, variational schemes, etc.).

We can recover Markov State Models by choosing disjoint sets S_i and setting

$$\phi_i(x) = \begin{cases} 1 \text{ for } x \text{ in } S_i \\ 0 \text{ otherwise.} \end{cases}$$

To see this, note that we can divide $\sum_{j=1}^{M} L_{ij} a_j = h_i - r_i$ by $\int \phi(x)\mu\,(dx)$ without changing the solution. Then,

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

where P is the MSM transition matrix and I is the identity matrix.

Diffusion map as a basis (equations)

However, we can choose other bases. In particular, we consider diffusion maps:

1. For each pair of datapoints (x_m, x_n), construct the kernel matrix
$$K_\varepsilon = \exp\left[-\frac{||x_m - x_n||^2}{2\varepsilon\sigma(x_m)\sigma(x_n)}\right]$$
where σ is a bandwidth function (Berry, Giannakis, and Harlim, 2015).
2. Normalize:
$$P_{mn}^{\text{DMAP}} = \frac{K_\varepsilon(x_m, x_n)}{\sum_n K_\varepsilon(x_m, x_n)},$$
3. Extract the submatrix with $x_m, x_n \in D$.
4. Set $\phi_i(x_m)$ to the eigenvectors of the submatrix with largest eigenvalues.

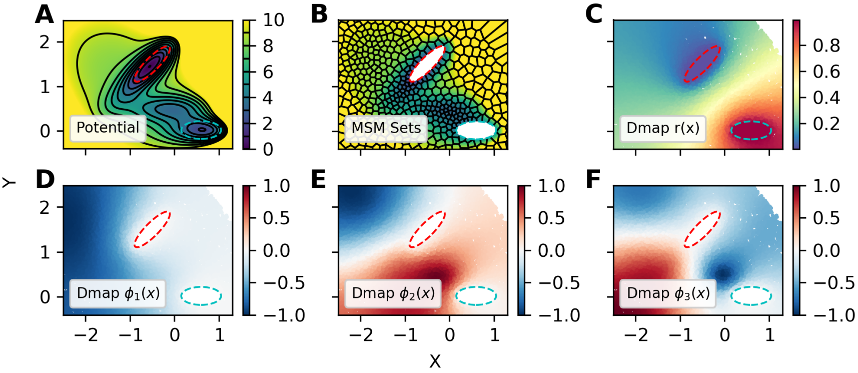
Our scheme is distinguished by its focus on obtaining dynamical statistics directly from operator equations.

With this in mind, let's now compare the performance of indicator (MSM) and diffusion map basis sets for solving such equations.

- Müller-Brown potential with “nuisance” degrees of freedom
- Fip35 protein folding

Comparison of basis sets (toy example)

We will use the Müller-Brown potential (U_{MB}) as an example for visualization.



MSM states are determined by *k*-means clustering, and there is no symmetrization of the transition matrix

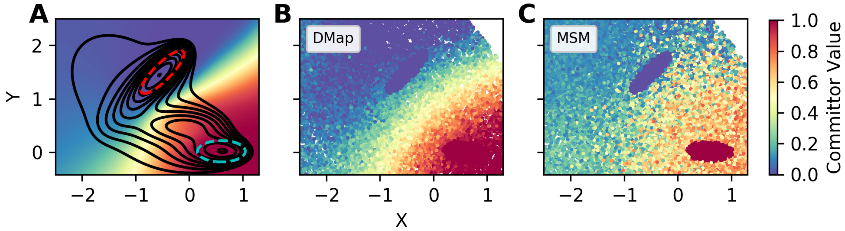
Comparison of basis sets (toy example)

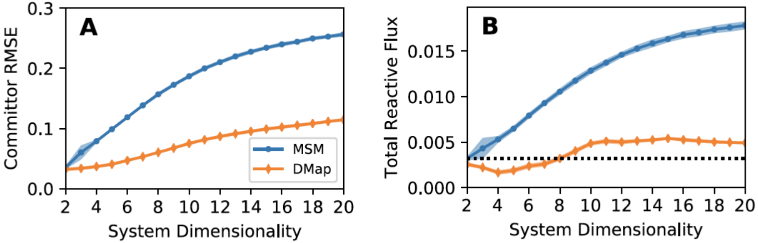
To artificially increase the dimensionality of the system, we include 18 additional harmonic degrees of freedom:

$$U\left(x, y, z_1, \ldots, z_{18}\right) = U_{\mathrm{MB}}(x, y) + \sum_{d=1}^{18} z_d^2.$$

- 10000 initial points drawn from the stationary distribution on the interval $x \in (-2.5, 1.5)$, $y \in (-2.5, 1.5)$.
- Each trajectory is 500 steps of overdamped Langevin dynamics with time step of 0.01 units (BAOAB integrator).
- The position is saved every 100 steps.

Comparison of committor estimates





- In each case, 500 basis functions are used.
- Averages of the RMS error are over 30 runs.

Treating memory

All forms of projection lead to memory.

In MSMs and related schemes, this memory is often addressed by increasing the lag time of the transition operator. The hope is that there is rapid averaging over projected degrees of freedom.

This effectively makes the approximation

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

This introduces a systematic bias in the answer that increases with s.

Treating memory

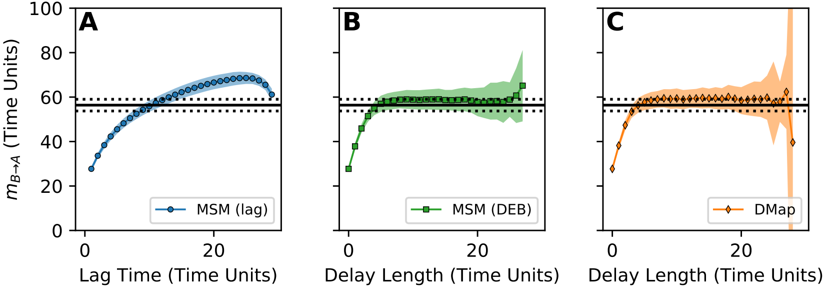
Our framework allows us to consider instead using delay embedding (Takens *et al.*, 1981) to include history information.

Let $\zeta^{(t)}$ be the projection of $\Xi^{(t)}$ at time *t*. We define the delay embedded process with *d* delays as

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

Comparison of methods for treating memory

- For the Müller-Brown example, use only the y coordinate.
- Redefine state A to be the set $\{y > 1.15\}$, and state B to be the set $\{y < 0.15\}$.
- 2000 trajectories, each sampled for 30000 time steps.

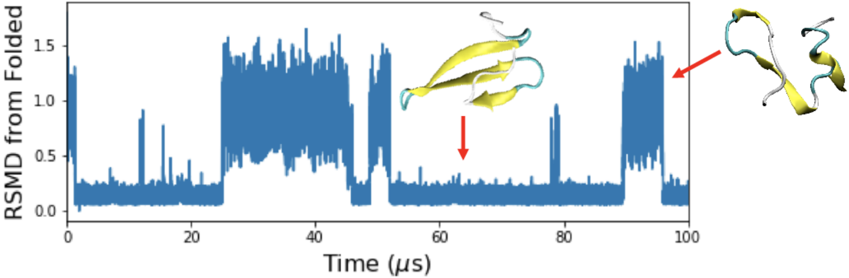


Increasing the lag time does not give a convergent estimate of the MFPT, in contrast to delay embedding.

Comparison of basis sets (protein example)

Fip35 WW domain trajectories generated by D.E. Shaw Research (Shaw *et al.*, 2010).

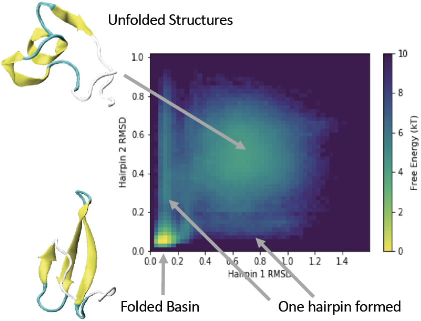
- 6 trajectories, each of length 100 μs with frames output every 0.2 ns.
- To reduce memory requirements, we subsample the trajectories, keeping every 100th frame.



Fip35 numerical details

There are two β-hairpins (β_1 and β_2) , defined as amino acids 7-23 and 18-29, respectively.

- CVs are pairwise distances between every other α-carbon (153 dimensions)
- Folded state:
  $\text{RMSD}_{\beta 1} < 0.2$ nm
  $\text{RMSD}_{\beta 2} < 0.13$ nm
- Unfolded state:
  $0.4 \text{ nm} < \text{RMSD}_{\beta 1} < 1.0$ nm
  $0.3 \text{ nm} < \text{RMSD}_{\beta 2} < 0.75$ nm



Unfolded Structures

Folded Basin

One hairpin formed

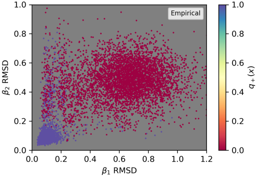
A scheme for evaluating performance

Because the data are not sufficient to estimate dynamical statistics without constructing a model, we split the data into training and test sets (half and half) and minimize costs:

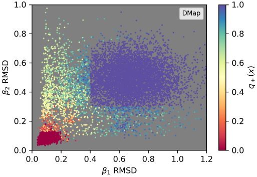
$$m_A(x) = \underset{f(x)}{\arg\min}\, \mathbf{E}\left[(\tau_A - f(x))^2\right]$$

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

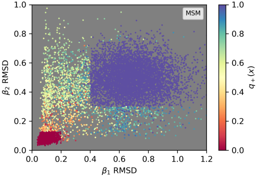
Input



Output (DMAP)

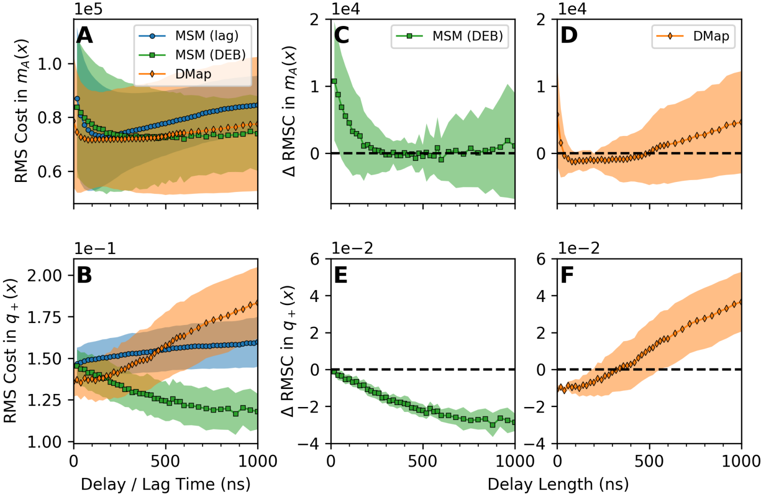


Output (MSM)



Fip35 test/train results

Results for 200 basis functions, differences are relative to MSM with the lag time.



Conclusions

We introduce a new framework for estimating dynamical averages based on the generator and the solution of associated equations by Galerkin approximation.

We have compared indicator (MSM) and diffusion map basis sets for selected problems.

Delay embedding provides a robust way of treating memory, in contrast to standard MSM schemes based on the lag time.

Thiede, Giannakis, Dinner, Weare, arXiv:1810.01841