1

2

Boltzmann priors for Implicit Transfer Operators

Juan Viguera Diez[1, 2] Mathias Schreiner[1] Ola Engkvist[1, 2] Simon Olsson[1]

@smnlssn
simonols@chalmers.se

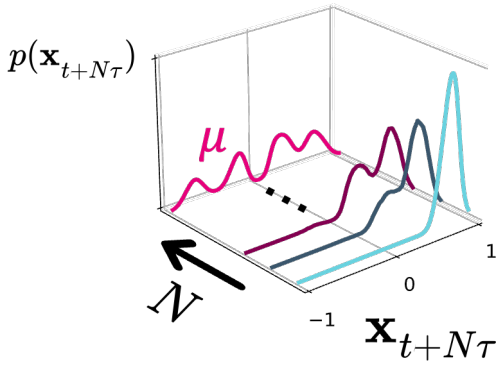
Molecular observables

- Predicting properties of molecules is important:
  - Drug design:
    - How well a drug binds to a target (affinity).
    - How long it stays bound (k_{off}).
  - Biophysics:
    - How stable the folded state of a protein is (ΔG).
    - How stability changes with mutations ($\Delta\Delta G$).

- Stationary: $O_a = \mathbb{E}_{\mu}\left[a(\mathbf{x})\right]$

- Dynamic: $O_{a(t),b(t+N\tau)} = \mathbb{E}_{\mathbf{x}_t \sim \mu}\left[\mathbb{E}_{\mathbf{x}_{t+N\tau} \sim p_\tau(\mathbf{x}_{t+N\tau}|\mathbf{x}_t)}\left[a(\mathbf{x}_t) \cdot b(\mathbf{x}_{t+N\tau})\right]\right]$

Computing molecular observables



The transition density, $p_\tau(\mathbf{x}_{t+N\tau}|\mathbf{x}_t)$, encodes the physical dynamics of a molecule.

The Boltzmann distribution,

$$\mu(\mathbf{x}) = \mathcal{Z}^{-1} \exp(-\beta U(\mathbf{x})), \text{ with } \mathcal{Z} = \int \mathrm{d}\mathbf{x} \exp(-\beta U(\mathbf{x})) \ ,$$

is the equilibrium distribution of the molecular dynamics.

Limitations:

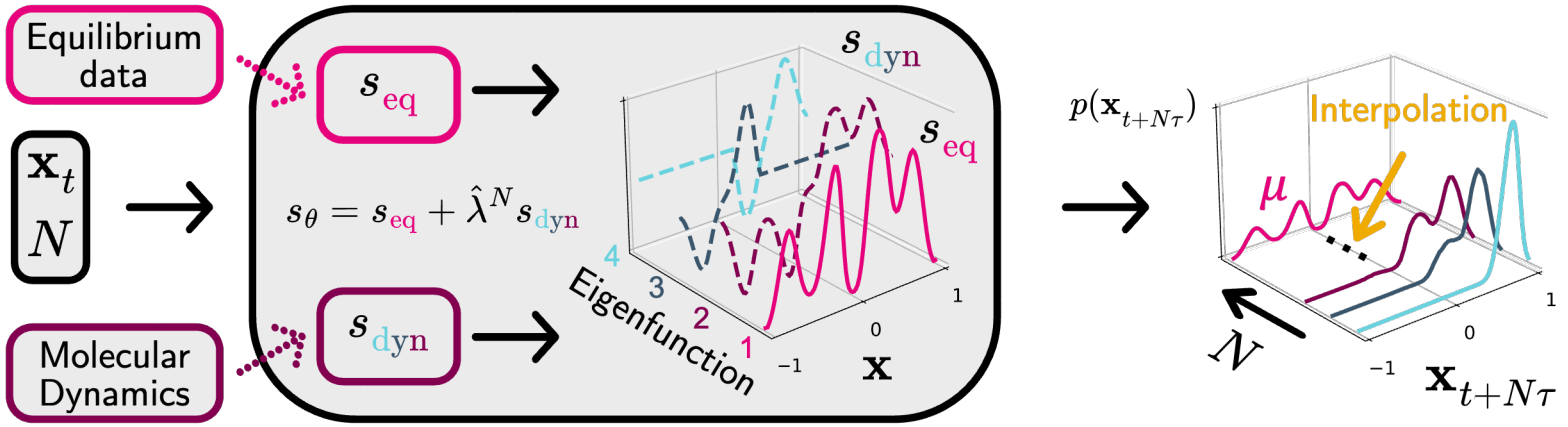
- Integration time steps required for stable simulations are tiny compared to time-scales of interest.
- Generating i.i.d. samples from the Boltzmann distribution is challenging.

Surrogate models of molecular dynamics

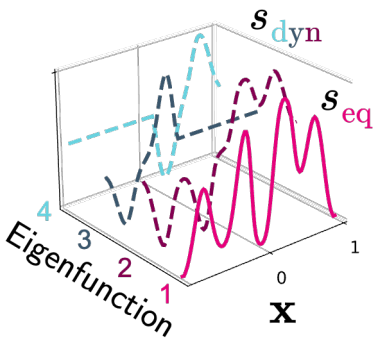
- Promising methods to efficiently compute molecular observables consist of learning surrogate models of key distributions:
  - Boltzmann Generators [1] are surrogates of the Boltzmann distribution.
  - Implicit Transfer Operators [2] are surrogates of the transition density.
- In this work, we build surrogate models under the Diffusion Denoising Probabilistic Models (DDPM) [3] framework.

[1] Noé F., Olsson S., Köhler J, Wu H. **Science** (2019). 365, eaaw1147. doi:10.1126/science.aaw1147
[2] Schreiner M., Winther O., Olsson S. **NeurIPS** (2023). doi: 10.48550/arXiv.2305.18046
[3] Ho J., Jain A., Abbeel P. **NeurIPS** (2020). doi: 10.48550/arXiv.2006.11239

Can we improve training of Implicit Transfer Operators using pretrained Boltzmann Generators?



Inductive bias



- Inductive bias: long-term dynamics is equilibrium.
- Eigen decomposition of the transition density:

$$p(\mathbf{x}_{N\tau}|\mathbf{x}_0) = \mu(\mathbf{x}_{N\tau}) + \sum_{i=2}^{\infty} \lambda_i^N(\tau)\psi_i(\mathbf{x}_0)\phi_i(\mathbf{x}_{N\tau}) \quad \text{with } 0 < \lambda_i < 1.$$

Equilibrium Non-equilibrium

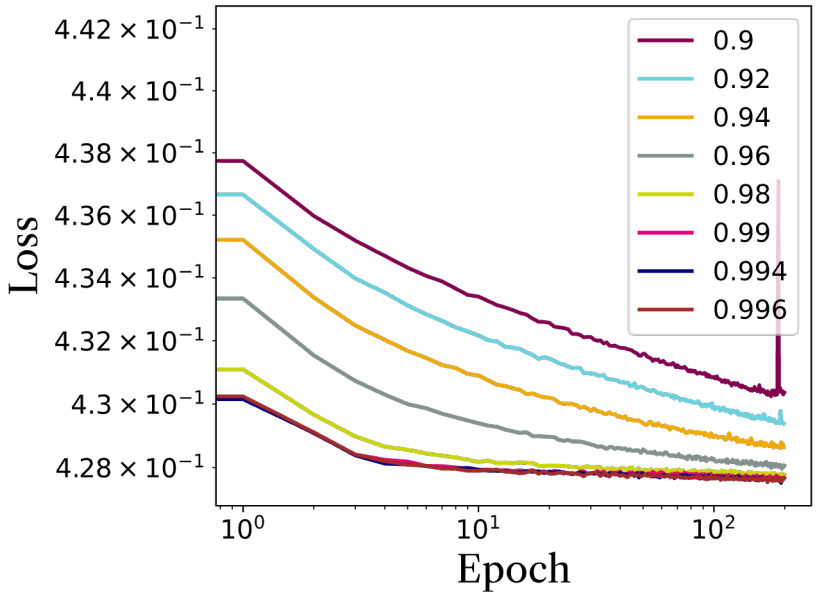
- Model:

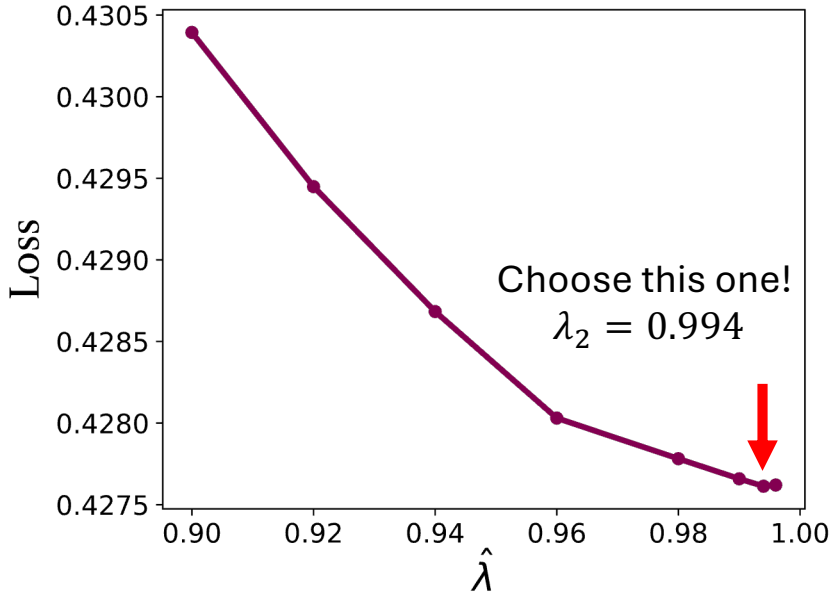
$$s(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}; \boldsymbol{\theta}) = s_{\mathrm{eq}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}}) + \hat{\lambda}^N s_{\mathrm{dyn}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}; \boldsymbol{\theta}) \quad \text{with } 0 < \hat{\lambda} < 1.$$

- Training: Score matching.
  - $s_{\mathrm{eq}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, t_{\mathrm{diff}})$ is a pretrained Boltzmann Generator (i.e. trained on enhanced sampling data).
  - $s_{\mathrm{dyn}}(\mathbf{x}_{N\tau}^{t_{\mathrm{diff}}}, \mathbf{x}_0, N, t_{\mathrm{diff}}; \boldsymbol{\theta})$ is trained with unbiased molecular dynamics data.
  - $\hat{\lambda}^N$ is a hyper-parameter defining a global relaxation time for the dynamics.

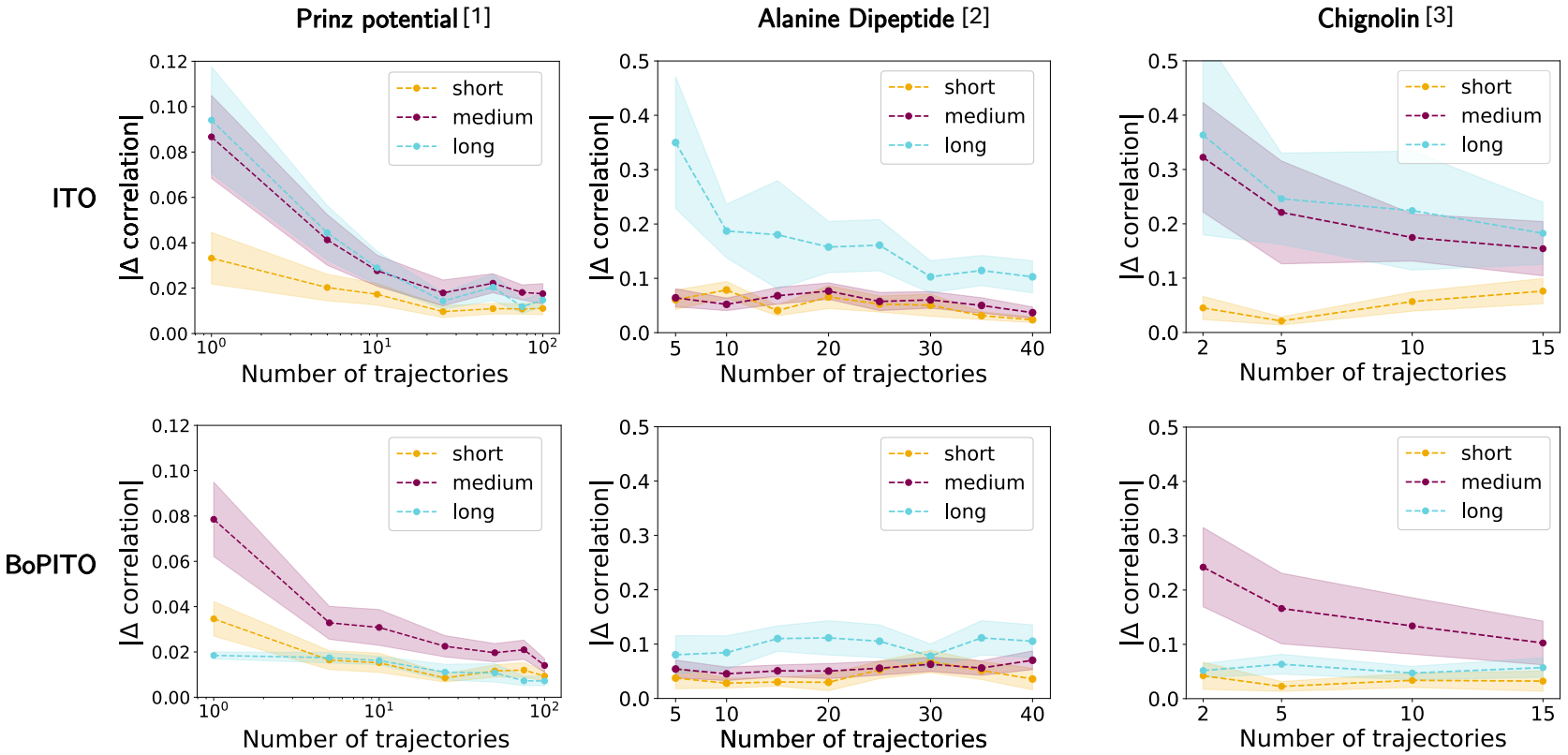
Fitting $\hat{\lambda}$

- When $\hat{\lambda}$ is too small, the system is prematurely forced to relax to equilibrium, leading to sub-optimal learning.
- Increasing $\hat{\lambda}$ improves learning. We choose the $\hat{\lambda}$ for which the loss stops decreasing ($\hat{\lambda} \sim \lambda_2$) .



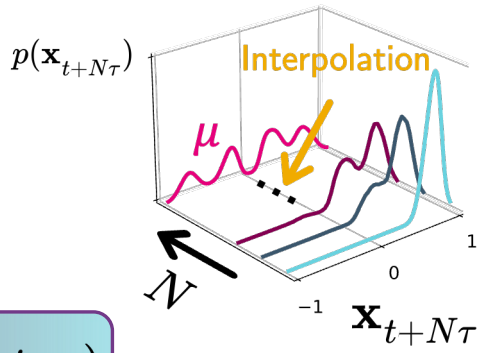


Error in correlation functions



[1] Prinz J.H., Wu H., Sarich M., Keller B., Senne M., Held M., Chodera J.D., Schütte C., Noé F. **J Chem Phys** (2011) doi: 10.1063/1.3565032
[2] Dibak M., Klein L., Krämer A., Noé F. **Phys. Rev. Res** (2022). doi: 10.48550/arXiv.2108.01590
[3] Lindorff-Larsen K., Piana S., Dror R.O., Shaw D.E. **Science** 2011. doi: 10.1126/science.1208351.

Interpolating between distributions



- Normal sampling, $N \leq N_{max}$ (max training lag):

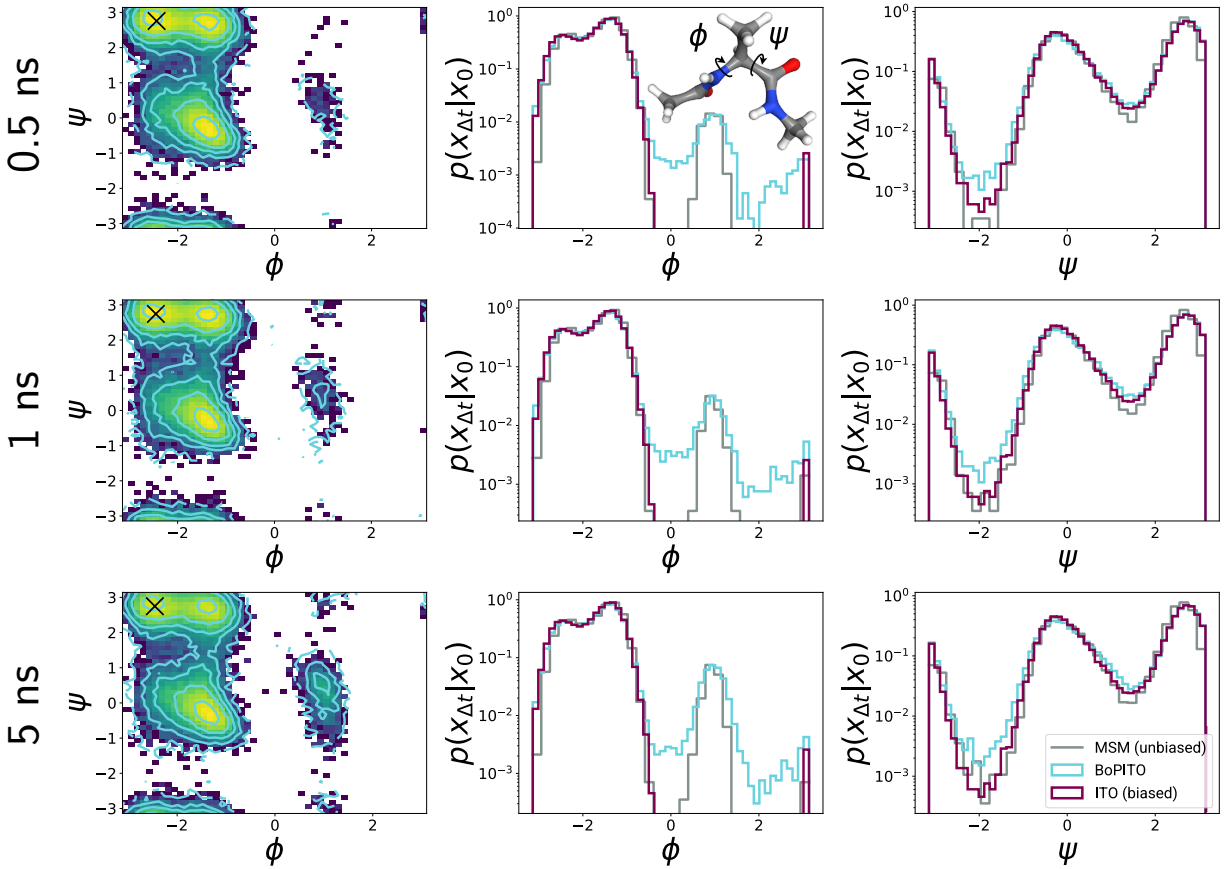
$$s(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}}) = s_{\text{eq}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^{N} s_{\text{dyn}}(\mathbf{x}_{N\tau}^{t_{\text{diff}}}, \mathbf{x}_0, N, t_{\text{diff}})$$

- Interpolation, $N_{int} > N_{max}$:

$$s(\mathbf{x}^{t_{\text{diff}}}, \mathbf{x}_0, N_{\text{int}}, t_{\text{diff}}) = s_{\text{eq}}(\mathbf{x}^{t_{\text{diff}}}, t_{\text{diff}}) + \hat{\lambda}^{N_{\text{int}}} s_{\text{dyn}}(\mathbf{x}^{t_{\text{diff}}}, \mathbf{x}_0, N_{\text{max}}, t_{\text{diff}})$$

- Multi-modal strategy: Select most consistent ensemble with unbiased dynamic observables (such as experimental data): Choose N_{int} that minimizes $|O_N^* - O_{N_{int}}|$.
- Use-case: Compensating for biases in the training trajectories.

Interpolation on Alanine Dipeptide



CHALMERS UNIVERSITY OF TECHNOLOGY [1]

AstraZeneca [2]

Thank you for listening!

Juan Viguera Diez[1, 2] Mathias Schreiner[1] Ola Engkvist[1, 2] Simon Olsson[1]

ICLR

WASP WALLENBERG AI, AUTONOMOUS SYSTEMS AND SOFTWARE PROGRAM

@smnlssn
simonols@chalmers.se