Chemical Science

ROYAL SOCIETY OF CHEMISTRY

EDGE ARTICLE

View Article Online
View Journal | View Issue

Check for updates

Cite this: *Chem. Sci.*, 2020, **11**, 9459

All publication charges for this article have been paid for by the Royal Society of Chemistry

Received 1st July 2020
Accepted 20th August 2020

DOI: 10.1039/d0sc03635h

rsc.li/chemical-science

Molecular latent space simulators

Hythem Sidky,[a] Wei Chen[b] and Andrew L. Ferguson *[a]

Small integration time steps limit molecular dynamics (MD) simulations to millisecond time scales. Markov state models (MSMs) and equation-free approaches learn low-dimensional kinetic models from MD simulation data by performing configurational or dynamical coarse-graining of the state space. The learned kinetic models enable the efficient generation of dynamical trajectories over vastly longer time scales than are accessible by MD, but the discretization of configurational space and/or absence of a means to reconstruct molecular configurations precludes the generation of continuous atomistic molecular trajectories. We propose latent space simulators (LSS) to learn kinetic models for continuous atomistic simulation trajectories by training three deep learning networks to (i) learn the slow collective variables of the molecular system, (ii) propagate the system dynamics within this slow latent space, and (iii) generatively reconstruct molecular configurations. We demonstrate the approach in an application to Trp-cage miniprotein to produce novel ultra-long synthetic folding trajectories that accurately reproduce atomistic molecular structure, thermodynamics, and kinetics at six orders of magnitude lower cost than MD. The dramatically lower cost of trajectory generation enables greatly improved sampling and greatly reduced statistical uncertainties in estimated thermodynamic averages and kinetic rates.

1 Introduction

Molecular dynamics (MD) simulates the dynamical evolution of molecular systems by numerically integrating the classical equations of motion.[1] Modern computer hardware[2,3] and efficient and scalable simulation algorithms[4–7] have enabled the simulation of billion[8,9] and trillion-atom systems.[10] Advancing the barrier in time scale has proven far more challenging. Stability of the numerical integration requires time steps on the order of femtoseconds commensurate with the fastest atomic motions,[11] which limits simulations to microseconds on commodity processors[11] and milliseconds on special purpose hardware.[3] Enhanced sampling techniques apply accelerating biases and analytical corrections to recover thermodynamic averages[12–16] but – except in special cases and the limit of small bias[17,18] – no analogous approaches exist to recover unbiased dynamical trajectories from biased simulations.

The MD algorithm propagates a molecular configuration $\mathbf{x}_t$ at time t to $\mathbf{x}_{t+\tau}$ *via* transition densities $\mathbf{x}_{t+\tau} \sim p_\tau(\mathbf{x}_{t+\tau}|\mathbf{x}_t)$.[19,20] Assuming ergodicity, the probability density over microstates converges to the stationary distribution as $\lim_{t\to\infty} q_t(\mathbf{x}) = \pi(\mathbf{x})$. Breaking the time scale barrier requires a surrogate model for $p_\tau(\mathbf{x}_{t+\tau}|\mathbf{x}_t)$ that can be more efficiently evaluated and with larger time steps than MD. Accurately approximating this propagator in the high-dimensional N-atom configurational space $\mathbf{x} \in \mathbb{R}^{3N}$ is intractable. In general, for sufficiently large τ there is an emergent low-dimensional simplicity that admits accurate modeling of the dynamics by a low-dimensional propagator $p_\tau(\psi_{t+\tau}|\psi_t)$ within a latent space $\psi \in \mathbb{R}^{m \ll 3N}$. The relation between MD and latent space dynamics can be represented as,[19,20]

$$\begin{tikzcd} \mathbf{x}_t \arrow[r, "E"'] \arrow[d, "\mathrm{MD}"'] & \psi_t \arrow[d, "P"] \\ \mathbf{x}_{t+\tau} & \psi_{t+\tau} \arrow[l, "D"] \end{tikzcd} \tag{1}$$

This scheme defines three learning problems:[19] (i) encoding E of molecular configurations $\mathbf{x}$ to the latent space ψ, (ii) propagation P of the latent space dynamics according to transition densities $p_\tau(\psi_{t+\tau}|\psi_t)$, and (iii) decoding (or generating) D of molecular configurations from the latent space.[19]

Markov state models (MSM)[21–28] and the equation-free approach of Kevrekidis and co-workers[29–35] respectively employ configurational and dynamical coarse graining to parameterize low-dimensional propagators, but both methods lack molecular decoders. Recently, numerous deep learning approaches have been proposed to learn E, P, and D from MD trajectories, including time-lagged autoencoders,[36] time-lagged variational autoencoders,[37] and time-lagged autoencoders with propagators.[38] Training these networks requires a time-lagged reconstruction term $\|\mathbf{x}_{t+\tau} - D \circ P \circ E(\mathbf{x}_t)\|$ within the loss, which can cause the network to fail to approximate the true slow modes.[39] Further, time-lagged autoencoders and time-lagged variational autoencoders do not learn valid propagators,[19] and the inherent

[a] Pritzker School of Molecular Engineering, University of Chicago, Chicago, USA. E-mail: andrewferguson@uchicago.edu

[b] Department of Physics, University of Illinois at Urbana-Champaign, Urbana, USA

stochasticity of MD appears to frustrate learning of the propagator and decoder in time-lagged autoencoders with propagators.[19] Deep generative MSMs (DeepGenMSM) simultaneously learn a fuzzy encoding to metastable states and "landing probabilities" to decode molecular configurations.[40] The method computes a proper propagator and generatively decodes novel molecular structures, but – as with all MSM-based approaches – it configurationally discretizes the latent space and relies on the definition of long-lived metastable states.

In this work, we propose molecular latent space simulators (LSS) as a means to train kinetic models over limited MD simulation data that are capable of producing novel atomistic molecular trajectories at orders of magnitude lower cost. The LSS can be conceived as means to augment conventional MD by distilling a kinetic model from training data, efficiently generating continuous atomistic trajectories, and computing high-precision estimates of any atomistic structural, thermodynamic, or kinetic observable.

The LSS is based on three deep learning networks independently trained to (i) learn an encoding E into a latent space of slow variables using state-free reversible VAMPnets (SRV),[41] (ii) learn a propagator P to evolve the system dynamics within this latent space using mixture density networks (MDN),[42,43] and (iii) learn a decoding D from the latent space to molecular configurations using a conditional Wasserstein generative adversarial network (cWGAN).[44] Separation of the learning problems in this manner makes training and deployment of the LSS modular and simple. The stochastic nature of the MDN propagator means that the trained kinetic model generates novel trajectories and does not simply recapitulate copies of the training data. The approach is distinguished from MSM-based approaches in that it requires no discretization into metastable states.[20,40] The continuous formulation of the propagator in the slow latent space shares commonalities with the equation-free approach,[29,31] but we eschew parameterizing a stochastic differential equation in favor of a simple and efficient deep learning approach, and also equip our simulator with a generative molecular decoder.

2 Methods

A schematic diagram of the LSS and the three deep networks of which it is comprised is presented in Fig. 1. We describe each of the three component networks in turn and then describe LSS training and deployment.

2.1 Encoder: state-free reversible VAMPnets

The transfer operator $\mathscr{T}$ at a lag time τ is the propagator of probability distributions over microstates with respect to the equilibrium density $u(\mathbf{x}) = q(\mathbf{x})/\pi(\mathbf{x})$ under transition densities $p_\tau(\mathbf{x}_{t+\tau}|\mathbf{x}_t)$.[45,46] For sufficiently large τ the dynamics may be approximated as Markovian so $p_\tau(\mathbf{x}_{t+\tau}|\mathbf{x}_t)$ is time homogenous,

$$u_{t+\tau}(\mathbf{x}) = \mathscr{T} \circ u_t(\mathbf{x}) = \frac{1}{\pi(\mathbf{x})} \int \mathrm{d}\mathbf{y} p_\tau(\mathbf{x}|\mathbf{y}) u_t(\mathbf{y}) \pi(\mathbf{y}). \tag{2}$$

In equilibrium systems obeying detailed balance $\pi(\mathbf{x})p_\tau(\mathbf{y}|\mathbf{x}) = \pi(\mathbf{y})p_\tau(\mathbf{x}|\mathbf{y})$, $\mathscr{T}$ is identical to the Koopman operator, self-adjoint with respect to $\langle a|b\rangle_\pi = \int a(\mathbf{x})b(\mathbf{x})\pi(\mathbf{x})\mathrm{d}\mathbf{x}$, and possesses a complete orthonormal set of eigenfunctions $\{\psi_i(\mathbf{x})\}$ with real eigenvalues $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \ldots,$[41,46–49]

$$\mathscr{T} \circ \psi_i(\mathbf{x}) = \lambda_i \psi_i(\mathbf{x}), \quad \langle \psi_i | \psi_j \rangle_\pi = \delta_{ij}. \tag{3}$$

The pair $(\psi_0(\mathbf{x}) = 1, \lambda_0 = 1)$ corresponds to the equilibrium distribution at $t \to \infty$ and the remainder to a hierarchy of increasingly quicker relaxing processes with implied time scales $t_i = -\tau/\ln \lambda_i$.[41] The evolution of $u_t(\mathbf{x})$ after k applications of $\mathscr{T}$ is expressed in this basis as,

$$u_{t+k\tau}(\mathbf{x}) = \mathscr{T}^k \circ u_t(\mathbf{x}) = \sum_i \langle \psi_i | u_t \rangle_\pi \exp\left(-\frac{k\tau}{t_i}\right) \psi_i(\mathbf{x}), \tag{4}$$

The variational approach to conformational dynamics (VAC) defines a variational principle to approximate these

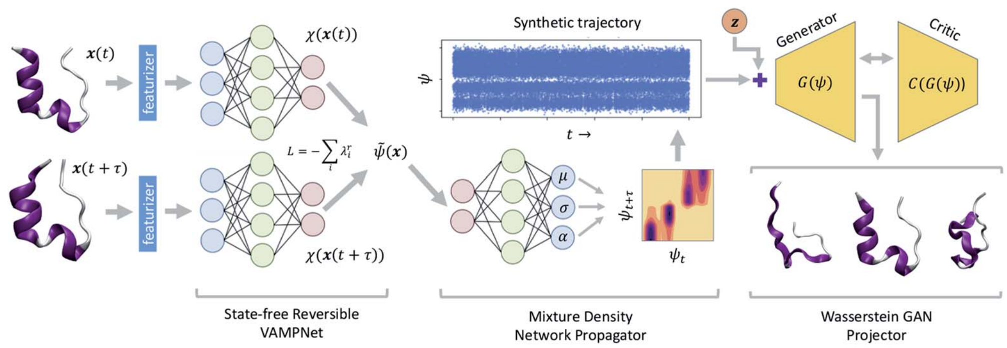

Fig. 1 Schematic diagram of the latent space simulator (LSS) comprising three back-to-back deep neural networks. A state-free reversible VAMPnet (SRV) learns an encoding E of molecular configurations into a latent space spanned by the leading eigenfunctions of the transfer operator (eqn (1)). A mixture density network (MDN) learns a propagator P to sample transition probabilities $p_\tau(\psi_{t+\tau}|\psi_t)$ within the latent space. A conditional Wasserstein GAN (cWGAN) learns a generative decoding D of molecular configurations conditioned on the latent space coordinates. The trained LSS is used to generate ultra-long synthetic trajectories by projecting the initial configuration into the latent space using the SRV, sampling from the MDN to generate long latent space trajectories, and decoding to molecular configurations using the cWGAN.

eigenfunctions as $\tilde{\psi}_i(\mathbf{x}) = \sum_{j} s_{ij}\chi_j(\mathbf{x})$ within a basis $\{\chi_j\}$ by solving for optimal expansion coefficients s_{ij}.[41,47,48] SRVs[41] – themselves based on VAMPnets, a deep learning-based method for MSM construction,[27] and closely related to extended dynamic mode decomposition with dictionary learning[50] – employ deep canonical correlation analysis (DCCA)[51] to learn both the optimal expansion coefficients and optimal basis functions as nonlinear transformations of the (featurized) molecular coordinates. This is achieved by training twin-lobed deep neural networks to minimize a VAMP-r loss function $\mathscr{L}_{\text{SRV}} = -\sum_{m} \lambda_m^{\text{r}}$.[27] SRVs trained over MD trajectories furnish an encoding E (eqn (1)) into a m-dimensional latent space spanned by $\{\psi_i(\mathbf{x})\}_{i=1}{}^{m}$, where m is determined by a gap in the eigenvalue spectrum. This spectral encoding into the leading modes of $\mathscr{T}$ neglects fast processes with implied timescales $t_i \ll \tau$ (eqn (4)) and is an optimal parameterization of the system for a low-dimensional long-time propagator.[19]

This slow subspace represents the optimal m dimensional embedding for the construction of the long-time propagator, but information is lost on more quickly relaxing processes. If there are faster processes in the system that are known to be of interest (*e.g.*, fast components of a folding transition), it may be possible to identify the higher order SRV eigenfunctions corresponding to these processes and judiciously expand the slow subspace to include these modes. Contrariwise, it can sometimes be the case that processes within the slow subspace (*e.g.*, slow *cis*/*trans* isomerizations) may not correspond to the kinetic process of interest (*e.g.*, folding). In certain situations it can be desirable to eliminate these slow "nuisance processes", perhaps because they correspond to very slow and rarely sampled events that are deleterious to constructing a robust kinetic model of the process of interest. One may eliminate these modes from the slow subspace *a posteriori* by simply discarding the SRV eigenfunctions identified to correspond to these processes, or – if they are known beforehand as a result of a previous analysis – *a priori* by subtracting them out of the system featurization using deflation.[52]

2.2 Propagator: mixture density networks

At sufficiently large τ the latent space $\psi(\mathbf{x}) = \{\psi_i(\mathbf{x})\}_{i=1}{}^{m}$ supports an autonomous dynamical system in the leading modes of $\mathscr{T}$. We train MDNs to learn transition densities $p_\tau(\psi_{t+\tau}|\psi_t)$ from MD trajectories projected in the latent space. MDNs combine deep neural networks with mixture density models to overcome poor performance of standard networks in learning multimodal distributions.[42,43] Transition densities are approximated as a linear combination of C kernels,

$$p_\tau(\psi_{t+\tau}|\psi_t) = \sum_{c=1}^{C} \alpha_c(\psi_t)\phi_c(\psi_{t+\tau}; \mu_c(\psi_t), \sigma_c(\psi_t)), \tag{5}$$

where we choose ϕ_c to be m-dimensional Gaussians. The ψ_t-dependent Gaussian means $\mu_c(\psi_t)$, variances $\sigma_c(\psi_t)$, and linear mixing coefficients $\alpha_c(\psi_t)$ are learned by a deep feedforward neural network that minimizes the loss function $\mathscr{L}_{\text{MDN}} = -\sum_{\gamma} \ln p_\tau(\psi_{t+\tau}{}^{\gamma}|\psi_t{}^{\gamma})$, where γ indexes pairs of time-lagged training data observations. The normalization $\sum_{c=1}^{C} \alpha_c(\psi_t) = 1$ is enforced by softmax activations and the $\mu_c(\psi_t)$ bounded using sigmoid activations.

The trained MDN defines the latent space propagator P (eqn (1)) and we sample transition densities $p_\tau(\psi_{t+\tau}|\psi_t)$ to advance the system in time (Fig. 1). Propagation is conducted entirely within the latent space and does not require recurrent decoding and encoding to the molecular representations that can lead to accumulation of errors and numerical instability.[19,53] The transition densities are learned from the statistics of transitions in the training data and new trajectories are generated by sampling from these transition densities. These new trajectories therefore represent novel dynamical pathways over the latent space and are not simply recapitulations or approximate copies of those in the training data. Successful MDN training is contingent on the low-dimensional and Markovian nature of the latent space dynamics at large τ discovered by the SRVs.

2.3 Decoder: conditional Wasserstein GAN

Generative adversarial networks are a leading neural network architecture for generative modeling.[54] We employ a cWGAN[44,55] to decode from the latent space ψ to molecular configurations $\mathbf{x}$ by performing adversarial training between a generator $G(\mathbf{z})$ that outputs molecular configurations from inputs $\mathbf{z} \sim \mathscr{P}_z(\mathbf{z})$ and a critic $C(\mathbf{x})$ that evaluates the quality of a molecular configuration $\mathbf{x}$. The networks are jointly trained to minimize a loss function based on the Wasserstein (Earth mover's) distance,

$$\mathscr{L}_{\text{WGAN}} = \max_{w \in W} \mathbb{E}_{\mathbf{x} \sim \mathscr{P}_x}[C_w(\mathbf{x})] - \mathbb{E}_{\mathbf{z} \sim \mathscr{P}_z}[C_w(G(\mathbf{z}))], \tag{6}$$

where $\mathscr{P}_x(\mathbf{x})$ is the distribution over molecular configurations observed in the MD training trajectory and $\{C_w\}_{w \in W}$ is a family of K-Lipschitz functions enforced through a gradient penalty.[44,55] To generate molecular configurations consistent with particular states in the latent space we pass ψ as a conditioning variable to G and C[56] and drive the generator with d-dimensional Gaussian noise $\mathscr{P}_z(\mathbf{z}) = \mathscr{N}(0,1) \in \mathbb{R}^d$. The noise enables G to generate multiple molecular configurations consistent with each latent space location. We train the cWGAN over $(\mathbf{x}^\gamma, \psi^\gamma)$ pairs by encoding each frame γ of the MD training trajectory into the latent space using the SRV. The trained cWGAN decoder D (eqn (1)) generates molecular configurations from the latent space trajectory produced by the propagator (Fig. 1).

3 Results and discussion

3.1 4-Well potential

We validate the LSS in an application to a 1D four-well potential[23] $V(x) = 2(x^8 + 0.8e^{-80x^2} + 0.2e^{-80(x-0.5)^2} + 0.5e^{-40(x+0.5)^2})$ for which analytical solutions are available. In this simple 1D system we construct the propagator directly in $x = \psi \in \mathbb{R}^1$, so encoding and decoding are unnecessary and this test validates that the MDN can learn transition densities $p_\tau(x_{t+\tau}|x_t)$ to accurately reproduce the system thermodynamics and kinetics. We generate a 5×10^6 time step Brownian dynamics trajectory in a dimensionless gauge with diffusivity $D = k_BT = 1000$ and a time step $\Delta t = 0.001$.[57] A MDN was trained using Adam[58] with early stopping over the [0,1] scaled trajectory at a lag time of $\tau = 100$, with $C = 8$ Gaussian kernels, and two hidden layers of 100 neurons with ReLU activations.[59] The trained MDN was used to generate a 5×10^4 step trajectory of the same length as the training data. Analytical transition densities were computed by partitioning the domain into 100 equal bins and defining the probability of moving from bin i to bin j as $p(j|i) = C_i e^{-(V_j - V_i)}$ for $|i - j| \leq 1$, where V_i is the potential at the center of bin i and C_i normalizes the total transition probability of bin i.[41]

The Brownian dynamics and synthetic LSS stationary distributions are in quantitative agreement with the analytical solution for the stationary density (Fig. 2a) and show very similar kinetic behaviors in their transitions between the four metastable wells (Fig. 2b). This agreement is due to the excellent correspondence between the analytical and learned transition densities (Fig. 2c and d).

3.2 Trp-cage miniprotein

We train our LSS over the 208 μs all-atom simulation of the 20-residue TC10b K8A mutant of the Trp-cage mini-protein performed by D. E. Shaw Research.[60] Generation of these MD trajectories would require ~2.5 days (2 million CPU h) on the special purpose Anton-2 supercomputer or ~6 months on a commodity GPU card.[3]

The SRV encoder was trained over a featurization of the trajectory employing backbone and sidechain torsions and C_α pairwise distances as informative and roto-translationally invariant descriptors.[25] We preprocess and represent the atomistic simulation data to the SRV in this manner to eliminate trivial rigid translations or rotations that would otherwise contaminate the learned slow modes. There are many choices of rotationally and translationally invariant featurizations, but we have shown in previous work on Trp-cage that this choice

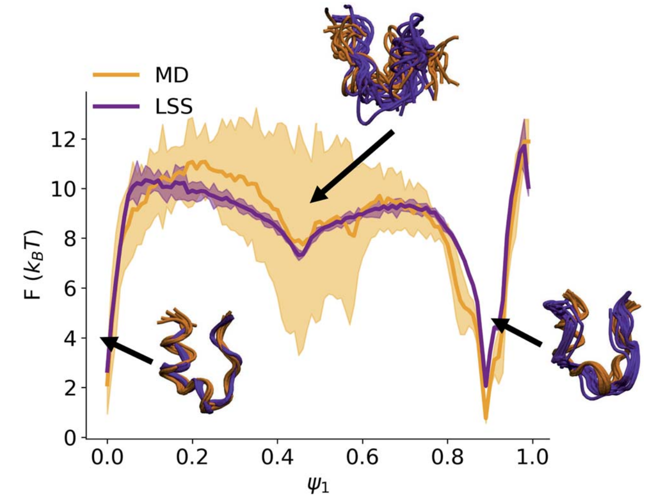

Fig. 3 Free energy profiles for the MD and LSS trajectories projected into the slowest latent space coordinate ψ_1. Shaded backgrounds represent standard errors estimated by five-fold block averaging. The profiles agree within a $0.91k_BT$ root mean squared error. Ten representative structures from the MD and LSS ensembles are sampled from the folded ($\psi_1 \approx 0$), unfolded ($\psi_1 \approx 0.9$), and metastable ($\psi_1 \approx 0.45$) regions.

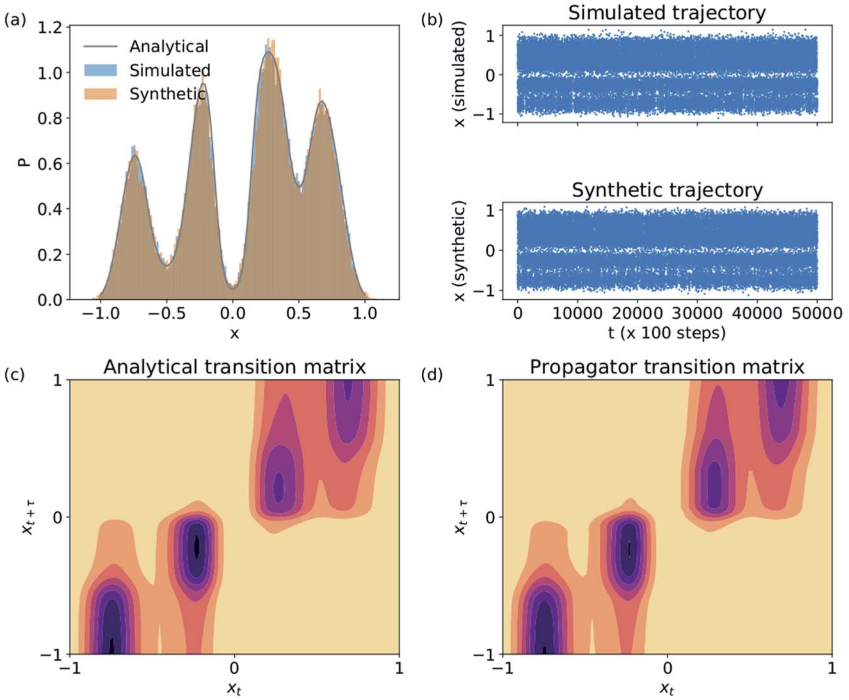

Fig. 2 Validation of the LSS in a 1D four-well potential. The MDN propagator predicts (a) a stationary distribution, (b) kinetic transitions, and (c and d) transition densities in excellent accord with analytical and Brownian dynamics results.

Table 1 Implied time scales of leading Trp-cage modes. Standard errors estimated by five-fold block averaging

Timescale	MD (μs)	LSS (μs)
t_1	3.00 ± 0.61	2.89 ± 0.12
t_2	0.54 ± 0.37	0.43 ± 0.04
t_3	0.45 ± 0.12	0.42 ± 0.01

contains more kinetic variance and generates more kinetically accurate models than either C_α pairwise distances alone, backbone and sidechain torsions alone, or rotationally and translationally aligned Cartesian coordinates.[25] We trained a SRV with two hidden layers with 100 neurons, tanh activations, and batch normalization using Adam[58] with a batch size of 200 000, learning rate of 0.01, and early stopping based on the validation VAMP-2 score.[25,41] A lag-time of $\tau = 20$ ns was chosen based on convergence of the transfer operator eigenvalues, and a $m = 3$-dimensional latent space encoding defined based on a gap in the eigenvalue spectrum. The MDN propagator was trained over the latent space projection of the MD trajectory at a lag time of $\tau = 20$ ns using Adam[58] with early stopping, $C = 24$ Gaussian kernels, and two hidden layers of 100 neurons with ReLU activations. The cWGAN decoder comprised a generator and critic with three hidden layers of 200 neurons with Swish[61] activations and a $d = 50$-dimensional noise vector. The training loss stabilized after 52 epochs. The cWGAN is trained to generate the Trp-cage C_α backbone by roto-translationally aligning MD training configurations to a reference structure. Training of the full LSS pipeline required ~1 GPU h on a NVIDIA GeForce GTX 1080 Ti GPU core.

The trained LSS was used to produce 100 × 208 μs synthetic trajectories each requiring ~5 s on a single NVIDIA GeForce GTX 1080 Ti GPU core. The LSS trajectories comprise the same total number of frames as the 208 μs all-atom trajectory but contain ~1070 folding/unfolding transitions compared to just

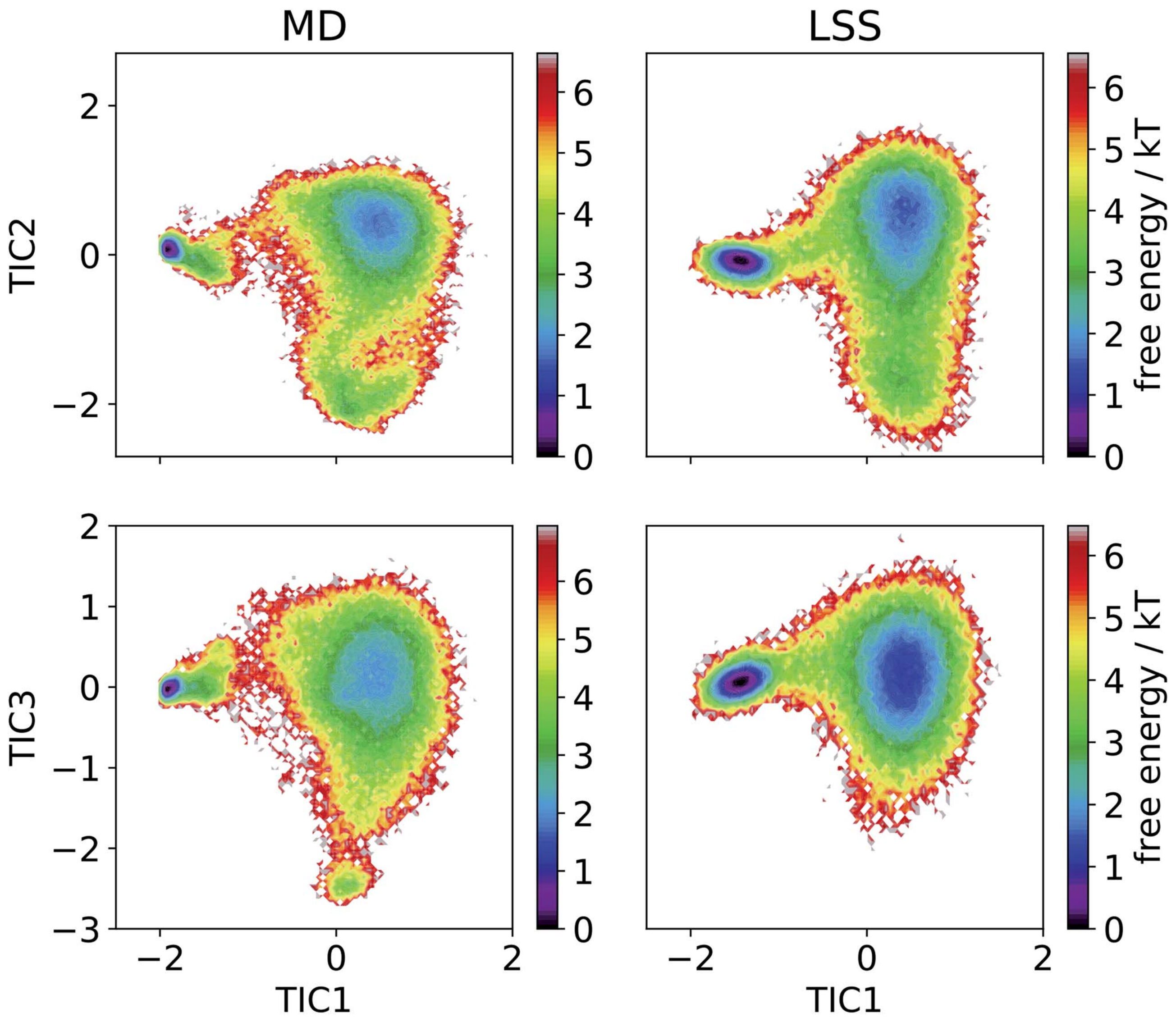

Fig. 4 Free energy profiles of the MD and LSS trajectories projected into the leading three MD TICA coordinates.

12 in the training data and were generated at six orders of magnitude lower cost. This observation illuminates the crux of the value of the approach: the LSS learns a kinetic model over limited MD training data and is then used to generate vastly longer novel atomistic trajectories that enable the observation of states and events that are only sparsely sampled in the training data. We now validate the thermodynamic, structural, and kinetic predictions of the LSS.

Thermodynamics. The free energy profiles projected into the slowest latent space coordinate $F(\psi_1) = -k_BT\ln(q(\psi_1))$ show excellent correspondence between the MD and LSS (Fig. 3). The free energy of the folded ($\psi_1 \approx 0$) and unfolded ($\psi_1 \approx 0.9$) basins and transition barrier are in quantitative agreement with a root mean squared error between the aligned profiles of $0.91k_BT$. The LSS profiles contain 10-fold lower statistical uncertainties than the MD over the same number of frames due to the 100-fold longer LSS data set enabled by their exceedingly low computational cost.

Structures. The MD and LSS molecular structures within the folded basin ($\psi_1 \approx 0$) and metastable transition state ($\psi_1 \approx 0.45$) possess a relative C_α-RMSD of 0.29 nm and 0.37 nm, respectively. Relative to the Trp-cage native state (PDB ID: 2JOF), the MD and LSS folded configurations possess a C_α-RMSD of 0.20 nm and 0.28 nm, respectively. The mean and standard deviation of the time-averaged radii of gyration (R_g) for the MD (0.87 ± 0.16) nm and LSS (0.87 ± 0.13) nm trajectories are indistinguishable with standard errors computed by five-fold block averaging. These results demonstrate that the LSS molecular structures are in excellent accord with MD.

Kinetics. We compare the MD and LSS kinetics through the autocorrelation times corresponding to the relaxation time scales associated with the $m = 3$ leading kinetic processes. The autocorrelation $C_i(\tau) = \langle\psi_i(t+\tau)\psi_i(t)\rangle/\langle\psi_i(t)\psi_i(t)\rangle$ of each process ψ_i is computed as an average over the MD data by projecting the training trajectory into the slow latent space through the trained SRV, whereas for the LSS data we use the latent space trajectory produced by the MDN. We fit the autocorrelation functions with a decaying exponential $\hat{C}_i(\tau) = \exp(-\tau/t_i)$ (*cf.* eqn (4)) and report t_i as the implied relaxation time scale associated with $\psi_i(t)$. Standard errors σ_{t_i} in t_i are estimated by breaking the LSS trajectory into $K = 5$ contiguous blocks of equal length, computing t_i^k over each of the $k = 1\ldots K$ blocks, and estimating σ_{t_i} – assuming the block size to be large enough to assume independent and uncorrelated estimates – from the standard deviation computed over the K estimates $\{t_i^k\}_{k=1}{}^K$.[62] The standard error provides a estimate of the deviation of the estimated value of t_i from the true value. All three time scales are in excellent agreement and again the LSS uncertainties are approximately 10-fold lower than the MD (Table 1).

We then employ time-lagged independent component analysis (TICA)[46,63–69] to determine whether the LSS trajectory possesses the same slow (linear) subspace as the MD. We featurize the trajectories with pairwise C_α distances and perform TICA at a lag time of $\tau = 20$ ns. Projection of the free energy surfaces into the leading three MD TICA coordinates show that the leading kinetic variance in the MD data is quite accurately reproduced by the LSS (Fig. 4). The only substantive disagreement is absence in the LSS projection of a small high-free energy metastable state at (TIC1 ≈ 0, TIC3 ≈ −2.5) corresponding to configurations with Pro18 dihedral angles $\psi \approx (-50)^\circ$. These configurations are only transiently occupied due to rare Pro18 dihedral flips that occur only twice during the 208 μs MD trajectory and are not contained in the $m = 3$-dimensional latent space.

4 Conclusions

We have presented LSS as a method to learn efficient kinetic models by training three state-of-the-art deep learning networks over MD training data and then using the trained model to generate novel atomistic trajectories at six orders of magnitude lower cost. The spirit of the approach is similar to MSM-based and equation-free approaches that use limited MD training data to parameterize highly-efficient kinetic models that can then be used to generate dynamical trajectories over vastly longer time scales than are possible with conventional MD. In contrast to these approaches, the absence of any discretization of the configurational space and provisioning with a molecular decoder enables the LSS to produce continuous atomistic molecular trajectories. Importantly, the probabilistic and generative nature of the approach means that the generated molecular trajectories are novel and not simply a reproduction of the training data, and the statistics of these trajectories accurately reproduce the structural, thermodynamic and kinetic properties of the molecular system.

The dramatic reduction in the cost of trajectory generation opens a host of valuable possibilities: vastly improved sampling of configurational space and dynamical transitions enable estimates of thermodynamic averages and kinetic rates with exceedingly low statistical uncertainties; parameterization of kinetic models with modest training data enable the production of ultra-long trajectories on commodity computing hardware; representation of the kinetic model as the parameters of a trio of deep networks enables efficient sharing of a "simulator in a box" that can then be used for rapid on-demand trajectory generation. The properties of the trained kinetic model – the dimensionality of the slow latent space, the structural correspondence of the slow modes, and the transition probabilities of the propagator – also provide fundamental insight and understanding of the physical properties of the molecular system.

As with all data-driven approaches, the primary deficiency of the LSS approach is that the resulting kinetic models are not necessarily transferable to other conditions or systems and are subject to systematic errors due to approximations in the molecular force fields and incomplete sampling of the relevant configurational space in the training data. The latter issue means that although the generated LSS trajectories are – similar to MSM-based and equation-free approaches – largely interpolative. The stochastic nature of the MDN propagator and generative nature of the cWGAN generator means that we may anticipate local extrapolations beyond the exact training configurations.[40] There is no expectation, however, that the trained model will discover new metastable states or kinetic transitions, and certainly not do so with the correct thermodynamic weights or dynamical time scales.

The present work has demonstrated LSS in a data-rich training regime where the MD training data comprehensively samples configurational space. The next step is to establish an adaptive sampling paradigm – similar to that in MSM construction[22] and some enhanced sampling techniques[70–73] – to enable its application in a data-poor regime. The adaptive sampling approach interrogates the kinetic model to identify under-sampled states and transitions that contribute most to uncertainties in the model predictions (*i.e.*, "known unknowns") and initializes new MD simulations to collect additional training data in these regions. This interleaving of MD training data collection and model retraining can dramatically reduce the required quantity of training data.[22] Moreover, new simulations initialized in under-sampled regions may also occasionally be expected to transition into new configurational states not present in the initial training data (*i.e.*, "unknown unknowns").[72] Iterating this process until convergence can expand the range of the trained kinetic model to encompass the relevant configurational space and minimize the cost of training data collection.

Finally, we also envisage applications of the LSS approach beyond molecular simulation to other fields of dynamical modeling where stiff or multi-scale systems of ordinary or partial differential equations, or the presence of activated processes or rare events, introduces a separation of time scales between the integration time step and events of interest. For example, there may be profitable adaptations of the approach in dynamical modeling within such fields as cosmology, ecology, immunology, epidemiology, and climatology.

Availability

Codes implementing the three deep networks required by the LSS are available at https://github.com/hsidky/srv/tree/newgen/.

Conflicts of interest

A. L. F. is a consultant of Evozyne and a co-author of US Provisional Patents 62/853,919 and 62/900,420 and International Patent Application PCT/US2020/035206.

Acknowledgements

This work was supported by MICCoM (Midwest Center for Computational Materials), as part of the Computational Materials Science Program funded by the U.S. Department of Energy, Office of Science, Basic Energy Sciences, Materials Sciences and Engineering Division, and by the National Science Foundation under Grant No. CHE-1841805. H. S. acknowledges support from the National Science Foundation Molecular Software Sciences Institute (MolSSI) Software Fellows program (Grant No. ACI-1547580).[74,75] We are grateful to D. E. Shaw Research for sharing the Trp-cage simulation trajectories.

Notes and references

1 D. Frenkel and B. Smit, *Understanding Molecular Simulation: From algorithms to applications*, Academic Press, San Diego, 2002.

2 J. E. Stone, D. J. Hardy, I. S. Ufimtsev and K. Schulten, *J. Mol. Graphics Modell.*, 2010, **29**, 116–125.

3 D. E. Shaw, J. P. Grossman, J. A. Bank, B. Batson, J. A. Butts, J. C. Chao, M. M. Deneroff, R. O. Dror, A. Even, C. H. Fenton, A. Forte, J. Gagliardo, G. Gill, B. Greskamp, C. R. Ho, D. J. Ierardi, L. Iserovich, J. S. Kuskin, R. H. Larson, T. Layman, L. S. Lee, A. K. Lerer, C. Li, D. Killebrew, K. M. Mackenzie, S. Y. H. Mok, M. A. Moraes, R. Mueller, L. J. Nociolo, J. L. Peticolas, T. Quan, D. Ramot, J. K. Salmon, D. P. Scarpazza, U. Ben Schafer, N. Siddique, C. W. Snyder, J. Spengler, P. T. P. Tang, M. Theobald, H. Toma, B. Towles, B. Vitale, S. C. Wang and C. Young, *SC'14: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, 2014, pp. 41–53.

4 J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, L. Kale and K. Schulten, *J. Comput. Chem.*, 2005, **26**, 1781–1802.

5 E. Chow, C. A. Rendleman, K. J. Bowers, R. O. Dror, D. H. Hughes, J. Gullingsrud, F. D. Sacerdoti and D. E. Shaw, *Desmond performance on a cluster of multicore processors (DESRES/TR–2008-01)*, DE Shaw Research Technical Report, 2008.

6 J. Glaser, T. D. Nguyen, J. A. Anderson, P. Lui, F. Spiga, J. A. Millan, D. C. Morse and S. C. Glotzer, *Comput. Phys. Commun.*, 2015, **192**, 97–107.

7 S. Plimpton, *Fast parallel algorithms for short-range molecular dynamics (SAND-91-1144)*, Sandia National Laboratory Technical Report, 1993.

8 F. F. Abraham, R. Walkup, H. Gao, M. Duchaineau, T. D. De La Rubia and M. Seager, *Proc. Natl. Acad. Sci. U. S. A.*, 2002, **99**, 5783–5787.

9 F. F. Abraham, R. Walkup, H. Gao, M. Duchaineau, T. D. De La Rubia and M. Seager, *Proc. Natl. Acad. Sci. U. S. A.*, 2002, **99**, 5777–5782.

10 N. Tchipev, S. Seckler, M. Heinen, J. Vrabec, F. Gratl, M. Horsch, M. Bernreuther, C. W. Glass, C. Niethammer, N. Hammer, B. Krischok, M. Resch, D. Kranzlmüller, H. Hasse, H.-J. Bungartz and P. Neumann, *Int. J. High Perform. Comput. Appl.*, 2019, **33**, 838–854.

11 R. Elber, *J. Chem. Phys.*, 2016, **144**, 060901.

12 G. M. Torrie and J. P. Valleau, *J. Comput. Phys.*, 1977, **23**, 187–199.

13 I. R. McDonald and K. Singer, *J. Chem. Phys.*, 1967, **47**, 4766–4772.

14 C. Abrams and G. Bussi, *Entropy*, 2014, **16**, 163–199.

15 Y. Miao and J. A. McCammon, *Mol. Simul.*, 2016, **42**, 1046–1055.

16 H. Sidky, W. Chen and A. L. Ferguson, *Mol. Phys.*, 2020, **118**, 1–21.

17 J. D. Chodera, W. C. Swope, F. Noé, J.-H. Prinz, M. R. Shirts and V. S. Pande, *J. Chem. Phys.*, 2011, **134**, 06B612.

18 L. Donati and B. G. Keller, *J. Chem. Phys.*, 2018, **149**, 072335.

19 F. Noé, 2018, arXiv preprint arXiv:1812.07669, https://arxiv.org/abs/1812.07669v1.

20 A. Fernández, *Ann. Phys.*, 2020, **532**, 1–5.

21 B. E. Husic and V. S. Pande, *J. Am. Chem. Soc.*, 2018, **140**, 2386–2396.

22 V. S. Pande, K. Beauchamp and G. R. Bowman, *Methods*, 2010, **52**, 99–105.

23 J. H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, M. Held, J. D. Chodera, C. Schtte and F. Noé, *J. Chem. Phys.*, 2011, **134**, 174105.

24 G. R. Bowman, V. S. Pande and F. Noé, *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation*, Springer Science & Business Media, 2013, vol. 797.

25 H. Sidky, W. Chen and A. L. Ferguson, *J. Phys. Chem. B*, 2019, **123**, 7999–8009.

26 C. Wehmeyer, M. K. Scherer, T. Hempel, B. E. Husic, S. Olsson and F. Noé, *Living Journal of Computational Molecular Science*, 2019, **1**, 1–12.

27 A. Mardt, L. Pasquali, H. Wu and F. Noé, *Nat. Commun.*, 2018, **9**, 5.

28 H. Wu and F. Noé, *J. Nonlinear Sci.*, 2020, **30**, 23–66.

29 I. G. Kevrekidis, C. W. Gear, J. M. Hyman, P. G. Kevrekidis, O. Runborg and C. Theodoropoulos, *Commun. Math. Sci.*, 2003, **1**, 715–762.

30 I. G. Kevrekidis, C. W. Gear and G. Hummer, *AIChE J.*, 2004, **50**, 1346–1355.

31 I. G. Kevrekidis and G. Samaey, *Annu. Rev. Phys. Chem.*, 2009, **60**, 321–344.

32 H. Mori, *Prog. Theor. Phys.*, 1965, **33**, 423–455.

33 R. Zwanzig, *J. Stat. Phys.*, 1973, **9**, 215–220.

34 R. Zwanzig, *Nonequilibrium Statistical Mechanics*, Oxford University Press, Oxford, 2001.

35 H. Risken and T. Frank, *The Fokker-Planck Equation: Methods of Solution and Applications*, Springer Verlag, Berlin Heidelberg New York, 2nd edn, 2012.

36 C. Wehmeyer and F. Noé, *J. Chem. Phys.*, 2018, **148**, 241703.

37 C. X. Hernández, H. K. Wayment-Steele, M. M. Sultan, B. E. Husic and V. S. Pande, *Phys. Rev. E*, 2018, **97**, 1–12.

38 B. Lusch, J. N. Kutz and S. L. Brunton, *Nat. Commun.*, 2018, **9**, 4950.

39 W. Chen, H. Sidky and A. L. Ferguson, *J. Chem. Phys.*, 2019, **151**, 064123.

40 H. Wu, A. Mardt, L. Pasquali and F. Noe, *Adv. Neural Inf. Process. Syst.*, 2018, **31**, 3975–3984.

41 W. Chen, H. Sidky and A. L. Ferguson, *J. Chem. Phys.*, 2019, **150**, 214114.

42 C. M. Bishop, *Mixture Density Networks (NCRG/94/004)*, Aston University Technical Report, 1994.

43 C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, Berlin, 2006.

44 I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. Courville, *Adv. Neural Inf. Process. Syst.*, 2017, **30**, 5768–5778.

45 P. Koltai, H. Wu, F. Noé and C. Schütte, *Computation*, 2018, **6**, 22.

46 S. Klus, F. Nüske, P. Koltai, H. Wu, I. Kevrekidis, C. Schütte and F. Noé, *J. Nonlinear Sci.*, 2018, **28**, 985–1010.

47 F. Noé and F. Nüske, *Multiscale Model. Simul.*, 2013, **11**, 635–655.

48 F. Nüske, B. G. Keller, G. Pérez-Hernández, A. S. Mey and F. Noé, *J. Chem. Theory Comput.*, 2014, **10**, 1739–1752.

49 H. Wu and F. Noé, *J. Nonlinear Sci.*, 2020, **30**, 23–66.

50 Q. Li, F. Dietrich, E. M. Bollt and I. G. Kevrekidis, *Chaos*, 2017, **27**, 103111.

51 G. Andrew, R. Arora, J. Bilmes and K. Livescu, *Proceedings of the 30th International Conference on Machine Learning*, 2013, pp. 2284–2292.

52 B. E. Husic and F. Noé, *J. Chem. Phys.*, 2019, **151**, 054103.

53 J. Pathak, B. Hunt, M. Girvan, Z. Lu and E. Ott, *Phys. Rev. Lett.*, 2018, **120**, 24102.

54 I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville and Y. Bengio, 2014, arXiv preprint arXiv:1406.2661, https://arxiv.org/abs/1406.2661v1.

55 M. Arjovsky, S. Chintala and L. Bottou, *Proceedings of the 34th International Conference on Machine Learning*, 2017, pp. 298–321.

56 M. Mirza and S. Osindero, 2014, arXiv preprint arXiv:1411.1784, https://arxiv.org/abs/1411.1784v1.

57 K. A. Beauchamp, G. R. Bowman, T. J. Lane, L. Maibaum, I. S. Haque and V. S. Pande, *J. Chem. Theory Comput.*, 2011, **7**, 3412–3419.

58 D. P. Kingma and J. Ba, 2014, arXiv preprint arXiv:1412.6980, https://arxiv.org/abs/1412.6980v1.

59 I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press, Cambridge, MA, 2016.

60 K. Lindorff-Larsen, S. Piana, R. O. Dror and D. E. Shaw, *Science*, 2011, **334**, 517–520.

61 P. Ramachandran, B. Zoph and Q. V. Le, 2017, arXiv preprint arXiv:1710.05941, https://arxiv.org/abs/1710.05941v1.

62 A. Grossfield, P. N. Patrone, D. R. Roe, A. J. Schultz, D. W. Siderius and D. M. Zuckerman, *Living Journal of Computational Molecular Science*, 2018, **1**, 5067.

63 G. Pérez-Hernández, F. Paul, T. Giorgino, G. De Fabritiis and F. Noé, *J. Chem. Phys.*, 2013, **139**, 07B604_1.

64 F. Noé and F. Nuske, *Multiscale Model. Simul.*, 2013, **11**, 635–655.

65 F. Nüske, B. G. Keller, G. Pérez-Hernández, A. S. J. S. Mey and F. Noé, *J. Chem. Theory Comput.*, 2014, **10**, 1739–1752.

66 F. Noé and C. Clementi, *J. Chem. Theory Comput.*, 2015, **11**, 5002–5011.

67 F. Noé, R. Banisch and C. Clementi, *J. Chem. Theory Comput.*, 2016, **12**, 5620–5630.

68 G. Pérez-Hernández and F. Noé, *J. Chem. Theory Comput.*, 2016, **12**, 6118–6129.

69 C. R. Schwantes and V. S. Pande, *J. Chem. Theory Comput.*, 2013, **9**, 2000–2009.

70 W. Chen, A. R. Tan and A. L. Ferguson, *J. Chem. Phys.*, 2018, **149**, 072312.

71 E. Chiavazzo, R. Covino, R. R. Coifman, C. W. Gear, A. S. Georgiou, G. Hummer and I. G. Kevrekidis, *Proc. Natl. Acad. Sci. U. S. A.*, 2017, **114**, E5494–E5503.

72 J. Preto and C. Clementi, *Phys. Chem. Chem. Phys.*, 2014, **16**, 19181–19191.

73 W. Zheng, M. A. Rohrdanz and C. Clementi, *J. Phys. Chem. B*, 2013, **117**, 12769–12776.

74 A. Krylov, T. L. Windus, T. Barnes, E. Marin-Rimoldi, J. A. Nash, B. Pritchard, D. G. Smith, D. Altarawy, P. Saxe, C. Clementi, T. D. Crawford, R. J. Harrison, S. Jha, V. S. Pande and T. Head-Gordon, *J. Chem. Phys.*, 2018, **149**, 180901.

75 N. Wilkins-Diehr and T. D. Crawford, *Comput. Sci. Eng.*, 2018, **20**, 26–38.