npj | computational materials

Article

Published in partnership with the Shanghai Institute of Ceramics of the Chinese Academy of Sciences

https://doi.org/10.1038/s41524-026-02076-z

Machine learning the entropy to estimate free energy differences without sampling transitions

Yamin Ben-Shimon[1,2], Barak Hirshberg[2,3,4] ✉ & Yohai Bar-Sinai[1,2,5,6] ✉

Thermodynamic phase transitions, a central concept in physics and chemistry, are typically controlled by an interplay of enthalpic and entropic contributions. In most cases, the estimation of the enthalpy in simulations is straightforward but evaluating the entropy is notoriously hard. As a result, it is common to induce transitions between the metastable states and estimate their relative occupancies, from which the free energy difference can be inferred. However, for systems with large free energy barriers, sampling these transitions is a significant computational challenge. Dedicated enhanced sampling algorithms require significant prior knowledge of the slow modes governing the transition, which is typically unavailable. We present an alternative approach, which only uses short simulations of each phase separately. We achieve this by employing a recently developed deep learning model for estimating the entropy and hence the free energy of each metastable state. We benchmark our approach by calculating the free energies of crystalline and liquid metals. Our method features state-of-the-art precision in estimating the melting transition temperature in Na and Al without requiring any prior information or simulation of the transition pathway itself.

Estimating the relative stability of different phases that are separated by high free energy barriers is a challenging task. To compare between phases, simulations must ergodically sample the relevant regions of phase space. However, due to the high free energy barriers, transitions are rare and metastable states persist for long timescales. Consequently, standard simulation techniques cannot sample transitions within feasible computational times[1].

Various approaches were developed to enhance the sampling efficiency of rare transitions. Many of them, such as Metadynamics[2–6], Umbrella Sampling[7,8], Gaussian accelerated molecular dynamics[9], or on-the-fly probability enhanced sampling[10–13], bias the system to induce transitions. These methods rely heavily on the identification of suitable collective variables (CVs), which should represent the slow modes of the system. This limits their applicability, since identifying appropriate CVs in condensed phases is challenging[14].

If, instead, we could estimate the entropy of each phase directly, without inducing transitions between the phases, we would be able to evaluate free energy differences using the thermodynamic relation

$$\Delta g = \Delta h - T\Delta s. \tag{1}$$

In Eq. (1), Δg is the free energy density difference, Δh is the enthalpy density difference, and Δs is the entropy density difference between the phases[15]. Note that this procedure is agnostic to the details of the transition path between metastable states. The challenge with this approach is that while calculating the enthalpy is straightforward, evaluating the entropy is generally an open problem: The enthalpy depends only on the energy, pressure and volume, which are routinely evaluated in simulations, but computing the entropy requires the partition function.

One line of work that estimates entropy from separate unbiased simulations is the 2PT method, which computes a dynamical density of states from the velocity autocorrelation function and decomposes it into diffusive and vibrational contributions[16]. However, the accuracy of the standard gas-solid partition can degrade for some liquids and dynamically unstable phases. By improving the treatment of the diffusive component,

[1]School of Physics, Tel Aviv University, Tel Aviv, Israel. [2]The Center for Physics and Chemistry of Living Systems, Tel Aviv University, Tel Aviv, Israel. [3]School of Chemistry, Tel Aviv University, Tel Aviv, Israel. [4]Center for Computational Molecular and Materials Science, Tel Aviv University, Tel Aviv, Israel. [5]Racah Institute of Physics, The Hebrew University of Jerusalem, Jerusalem, Israel. [6]The Rachel and Selim Benin School of Computer Science and Engineering, The Hebrew University of Jerusalem, Jerusalem, Israel. ✉e-mail: hirshb@tauex.tau.ac.il; ybarsinai@gmail.com

this work was also extended for liquid metals to obtain more accurate liquid entropy estimates[17].

A comprehensive review of methods for direct entropy estimation is beyond the scope of this paper, and we focus on select recent works. Avinery et al.[18] and Martiniani et al.[19] leveraged established lossless compression algorithms to bound the information content of a sequence of sampled microstates, which is equivalent to the thermodynamic entropy. The applicability of these methods depends on the performance of the underlying compression algorithm to the problem at hand[20]. Since compression algorithms treat data as a one-dimensional string, this approach works well for systems with a natural sequential structure, but faces difficulties when forcing more complicated geometries into a 1D sequence[21]. More recently, Sorkin et al. developed a novel upper bound on the entropy, based on correlations between various degrees of freedom[22,23]. These methods are promising, and successfully estimated the entropy in model systems, but were not applied to molecular simulations of phases separated by high free energy barriers.

Machine learning algorithms have also been successfully used to estimate the entropy in physical systems. Gelman et al. estimated the probability density of microstates directly by training an auto-regressive model[24]. Then, they calculated the entropy from the log-probability. However, this method is limited to lattice systems in two dimensions, and does not scale well with system size. Generative models have recently been proposed to directly sample from the Boltzmann distribution[25], with recent extensions advancing these approaches[26,27], where free energy, in principle, can be inferred from the relative frequencies of metastable states. Petersen et al. even used diffusion models to generate whole trajectories recently[28]. While promising, they have so far been applied mostly to small molecules[29]. Most relevant to this work is a method called MICE by Nir et al.[30]. They estimated the entropy by mapping the problem to an iterative process of mutual information (MI) estimation at different length scales. The latter was done by representing the MI as an optimization problem, parameterized by a neural network, as proposed by Belghazi et al.[31].

We adopt the MICE approach and combine it with molecular dynamics (MD) simulations for the first time. This offers a new method to estimate the entropy and relative stability of phases separated by high free-energy barriers. Our computationally efficient approach for learning the entropy requires only short simulations of each phase separately. It offers an accurate calculation of the critical temperature of melting phase transitions without the need for prior knowledge of the system (CVs) and without sampling transitions. We demonstrate the usefulness of our approach in estimating the melting temperature of Na. Without any changes to the model architecture or hyperparameters, our approach also accurately estimates the melting temperature of Al.

Results

MICE framework

The central concept underlying MICE is that the entropy can be estimated as the sum of MI contributions at different length scales. Consider two random variables A and B. Their mutual information $MI(A, B)$ is defined as:

$$S(A, B) = S(A) + S(B) - MI(A, B), \tag{2}$$

where $S(A, B)$ is the entropy of the joint distribution and $S(A)$, $S(B)$ are the marginal entropies. MICE uses this relationship to replace the estimation of the total entropy with that of the entropies of the subsystems and their MI. This approach is useful because the computational complexity of entropy estimation is exponential in the system size. By breaking the system into smaller, more manageable parts, while keeping track of the MI between them, the overall computational demands are significantly reduced.

Consider a physical system X_0 of volume V_0, which is split into two equal parts of volume $V_1 = \frac{1}{2}V_0$. We treat X_0 as a random variable drawn from the Boltzmann distribution. If the system is translationally invariant (which is the case far from boundaries or with periodic boundary conditions), the two halves are statistically identical and can be both denoted as X_1. With this setup, Eq. (2) simplifies to:

$$S(X_0) = 2S(X_1) - MI(X_1), \tag{3}$$

where $MI(X_1)$ is the mutual information between the two subsystems X_1. This process can be repeated for increasingly smaller systems, resulting in

$$s(X_0) = s(X_m) - \frac{1}{2}\sum_{k=1}^{m}\frac{MI(X_k)}{V_k}, \tag{4}$$

where $s(X_k) = S(X_k)/V_k$ is the entropy density for a subsystem X_k with volume V_k, see Supplementary Information for details and derivation. For later use, we also denote the interface area between two neighboring X_k systems by A_k. Eq. (4) decomposes the entropy S into contributions from different length scales. The entropy density of the smallest subdivision, $s(X_m)$, is calculated directly by brute force enumeration, relying on symmetry considerations, see SI, "Direct estimation of X_m" for details. In our application, we continue the division until X_m is small enough such that it contains a single particle on average. Since the volume of X_k decreases exponentially with k, the required number of subdivisions is logarithmic in the system size.

MI estimation and system representation

To estimate $MI(X_k)$, we employ the Mutual Information Neural Estimator (MINE)[31], which we describe here briefly. The method relies on the representation of MI between two random variables X and Y as the supremum over all real functions $\mathcal{T}(X, Y)$[32],

$$MI(X, Y) = \sup_{\mathcal{T}}\left\{\mathbb{E}_{\mathbb{P}_{XY}}[\mathcal{T}(X, Y)] - \log\left(\mathbb{E}_{\mathbb{P}_X\otimes\mathbb{P}_Y}\left[e^{\mathcal{T}(X,Y)}\right]\right)\right\}. \tag{5}$$

Here, $\mathbb{P}_{XY}$ is the joint probability distribution of the two variables and $\mathbb{P}_X \otimes \mathbb{P}_Y$ is the product of their marginal distributions. Then, MI is estimated by parameterizing $\mathcal{T}$ with a neural network, and optimizing its weights. Since there is no guarantee that the optimization will find the global supremum, this procedure bounds the true MI from below, resulting in an upper bound on the entropy (Eq. (4)).

To apply MICE, we first sample equilibrium snapshots for each phase by running separate, standard MD simulations. Our X_0 is an MD simulation box with periodic boundary conditions, see below for details. We obtain samples for each subdivision k from $\mathbb{P}_{X_kX_k}$ by choosing a volume V_k in a random location out of the original simulation box. Samples from $\mathbb{P}_{X_k} \otimes \mathbb{P}_{X_k}$ are generated by stitching together two halves of randomly chosen samples from the joint distribution. We then train a neural network to optimize Eq. (5). Figure 1 (left) shows representative training curves for a typical subdivision of solid and liquid Na. While the training curves are noisy[33], a running average shows convergence to different MI estimations for the two phases, which we take as the estimate for $MI(X_k)$. We repeat this procedure for $k = 1, \ldots, m$ and for each phase separately. The entropy of X_0 is obtained by summing over all subsystems through Eq. (4).

A key component of the method is the choice of representation of the configurations X_k for training. Many choices are possible, and here we adopt the representation used in ref. 30: a cubic grid with n^3 voxels, each of side length p. The value of a voxel is 1 if it contains at least one atom, and zero otherwise. As n increases, the embedding resolution increases, which we measure through the dimensionless number $R = \sigma/p$, where σ is the nearest-neighbor distance in the experimental crystal structure. Figure 1 (middle) shows that the MI converges for $R \approx 10$ for Na.

Another important point is that, since we are dealing with bulk properties, X_0 of Eq. (4) should be large enough to represent an effectively infinite system. The size of X_0 should thus be chosen such that it minimizes finite-size effects on the one hand, but is numerically manageable on the other hand. This "sweet spot" can be obtained by using the asymptotic area law of the entropy[34]: At length scales larger than the longest correlation

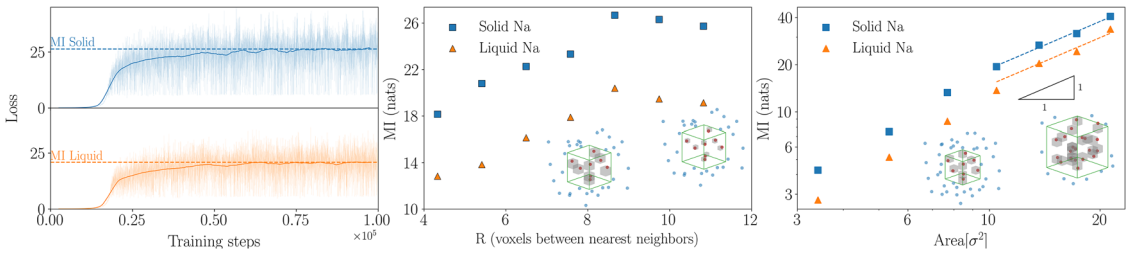

Fig. 1 | MI estimations for liquid and crystalline Na. (left) Training curves for solid (blue, upper plot) and liquid (orange, bottom plot) Na close to T_m. Bold lines represent running averages over 5000 steps and dashed lines are the converged values over last 10^4 steps. (middle) MI estimation for a system of fixed physical size as a function of the spatial resolution. The estimate plateaus at high resolutions. At very fine resolutions the voxelization is more faithful but requires substantially larger networks (for a fixed physical size). Under a fixed training budget, those larger models may not be trained as thoroughly, leading to a mild drop in MI estimation. (right) Estimation of MI between two halves of a cubic subsystem, as a function of the interface area. The dashed lines show a linear trendline, showing that large systems obey an area law. The corresponding figures for Al data are presented in SI.

length, $MI(X_k)$ should scale linearly with A_k. Indeed, Fig. 1 (right) shows MI as a function of A_k for increasing system size. We find that for small systems MI scales super-linearly with A_k, and becomes linear for larger systems, see dashed lines in Fig. 1 (right). We choose X_0 of Eq. (4) to be slightly larger than this crossover size. This allows us to take the bulk limit of infinitely large X_0 analytically, under the assumption that for systems larger than X_0, we have $MI(X_k) = \alpha A_k$, where α can be estimated from the linear fit in the right panel of Fig. 1. As shown in SI "Extrapolating MICE to bulk properties using the area law", if X_0 is large enough to be in the area law regime, this procedure results in

$$s = s(X_0) - 3\frac{\alpha A_0}{V_0}. \tag{6}$$

MI and entropy difference in Na

Figure 2 shows the MI per atom for solid and liquid Na for the different subsystems divisions of the MICE procedure (Eq. (4)). As expected, since structural information is encoded in long wavelengths, the difference in MI between the solid and liquid phases decreases for smaller subsystems with fewer atoms and vanishes for subsystems containing only one atom. The jump in MI per atom every three divisions corresponds to a divide with a large aspect ratio of interface area to volume (Fig. 2 (inset)).

With Eq. (4) and Eq. (6), these MI estimations are used to calculate the specific entropy density difference between the phases, Δs. Results are shown in Fig. 3a, c (labeled "MICE"), and compared to the estimates calculated from the same unbiased single-phase simulations using the first and the second order entropy expansions, s_{id} and s_2 (labeled "MD"). In addition, we show the WT-MetaD estimates which are obtained by fitting a linear trend to Δg values at different temperatures and using Eq. (1) (labeled "MetaD"). Enthalpy estimation in all three methods is also plotted, and is identical for MD and MICE simulations by definition, since they both use time averages of the same unbiased dynamics.

Unsurprisingly, it is seen that $\Delta s_{id} + \Delta s_2$ provides a poor estimation of the entropy difference. If combined with the measured enthalpy difference, it would predict a melting temperature of about 295 K, a relative error of ~20% in comparison to the experimental melting temperature. In our WT-MetaD calculations, the entropy is underestimated, but the resulting prediction of T_m lies within several kelvin of the experimental value, due to a cancellation of a similar error in Δh. This tendency was reported for crystallization transitions and is consistent with the known bias compression factor of WT-MetaD[3,35]. MICE, on the other hand, produces a modest overestimation in Δs, and an unbiased estimation of Δh, which yield a melting temperature closer to the experimental value for this benchmark.

Transferability to Al

After establishing that our method works reliably for the melting transition of Na, we applied exactly the same network, with identical hyperparameter settings (layer sizes, dropout, etc.), to a similar dataset for solid and liquid Al. This dataset was prepared with a similar voxel resolution and mean atomic density. Without any additional tuning or architectural changes, the model converged smoothly and delivered MI estimates, and consequently Δs estimates, which were used to predict the melting temperature, mirroring the accuracy obtained for Na, as shown in Fig. 3 b, d. This portability emphasizes the robustness and generality of our representation and training protocol. Here too, $\Delta s_{id} + \Delta s_2$ is a poor approximation of the entropy, and the WT-MetaD estimate of the melting temperature is consistent with a partial cancellation of errors in the enthalpy and entropy differences.

Discussion

To conclude, we present a machine-learning approach to estimate free-energy differences between metastable states that are separated by high barriers without sampling transitions between them. It does so by formulating the entropy difference as a sum over MI at different length scales. Then, the contribution of each length scale is obtained by optimizing a convolutional neural network. Our approach predicts the melting temperature of Na and Al using only short simulations of each phase separately, and without requiring collective variables or samples of the transition pathway.

While MICE shows state-of-the-art accuracy in entropy estimation for atomic systems, there is still much room for improvement. First, the voxel-based encoding may introduce spurious correlations and is not natural for molecular data: it neglects symmetries and local bonding, potentially distorting or omitting relevant information. This could be improved by adopting a graph-like representation with molecularly informed embeddings that better respect symmetry and local structure, as was done in a different context[36–38]. Second, the MINE estimator at the core of our approach[31] is optimized with stochastic, finite-batch gradients, which are biased and can inflate MI in the high-MI regime (see Supplementary Information for details). Using recently developed MINE variants that address bias may improve performance. These directions will be pursued in future research.

Methods

Molecular dynamics simulations

To benchmark our method, we performed well-tempered Metadynamics (WT-MetaD) simulations following the protocol of Piaggi et al.[35] using LAMMPS (15Jun2023)[39] and enhanced-sampling algorithms library PLUMED 2.8.3[40–42]. We ran NPT WT-MetaD simulations in the temperature ranges 300–400 K for Na and 850–950 K for Al, using their respective embedded atom models (EAM)[43,44]. We used the pair entropy s_2 and the enthalpy per atom as CVs. For Na, we deposited Gaussian hills with an initial height of 2.5 kJ mol^{-1} every 500 steps with widths of 0.2 kJ mol^{-1} (enthalpy per atom) and 0.1 k_B (s_2). For Al, we used a larger initial height of 7.5 kJ mol^{-1} and widths of 0.3 kJ mol^{-1} (enthalpy) and 0.1 k_B (s_2), keeping the same deposition stride. A bias factor of 30 was used for both systems.

Convergence was assessed as in ref. 35. We used a Parrinello-Rahman barostat at 1 bar acting on a triclinic simulation cell, allowing isotropic volume fluctuations while relaxing shear components to zero. Simulations were performed on a $5 \times 5 \times 5$ supercell for Na with 250 atoms and a $4 \times 4 \times 4$ supercell for Al with 256 atoms.

To generate data for MICE, we performed unbiased MD simulations of Na and Al separately in the solid (bcc and fcc, respectively) and liquid phases with LAMMPS. Collective variables were tracked using PLUMED 2.8.3[40–42], but not used for biasing the dynamics. Temperature was maintained at 365 K for Na and 900 K for Al, close to the experimental melting points, using a stochastic velocity-rescaling thermostat[45] with a relaxation time of 0.1 ps. Early exploratory tests indicated that, for a fixed subsystem size, the estimated MI exhibits only a weak dependence on temperature over the range considered (up to 30K away from the melting temperature).

The pressure was set to 1 bar using the isotropic Parrinello-Rahman barostat on an orthorhombic cell[46] with a relaxation time of 10 ps. Simulations were performed on an $8 \times 8 \times 8$ supercell with 1024 atoms for Na and 2048 atoms for Al. We ran 20 independent replicas with different seeds for each element and phase, and sampled configurations every 5 ps, for a total of 40000 data points, which were evenly split into training and validation sets.

Although MICE ultimately operates on smaller subsystems when estimating MI, we performed the MD simulations on larger supercells to mitigate finite-size and periodic-boundary artifacts. In particular, using larger cells reduces interactions of atoms with their periodic images, and finite-size biases to the phononic dispersion at long wavelengths. Larger supercells are also necessary for the MI extrapolation procedure, which requires MI estimates across multiple subsystem sizes. Since MICE only requires training data from short, unbiased simulations, the cost of increasing the supercell in comparison to WT-MetaD is minimal.

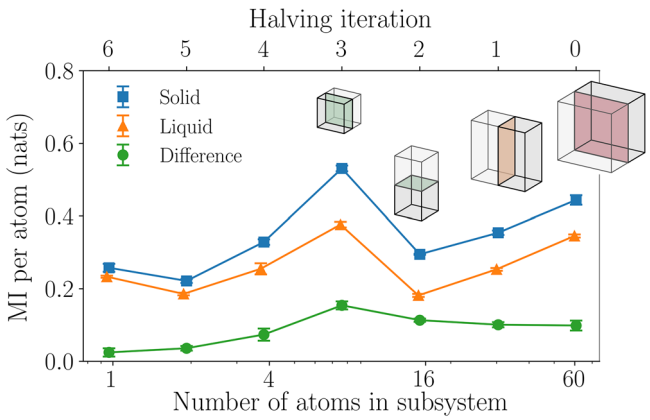

Fig. 2 | MI density across successive partitions. Starting from the rightmost panel, the system is halved along a different dimension at each step, and MI is computed across the resulting interface. Because the three cut directions cycle, the interface area is unchanged every third partition while the subsystem volume halves at every step (e.g., iterations 2-3). Consequently, the MI density doubles between such iterations.

Neural-network training

Training data was obtained by taking sub-volumes of the simulation box. X_0, the biggest subsystem used in Eq. (4), was taken to be a cubic volume containing, on average, 60 atoms for Na and 50 for Al, i.e., about a third of the linear size of the simulation box, to avoid periodic imaging effects. For each snapshot, a sub-volume was chosen at a random location and orientation, uniform over the simulation box and SO(3), respectively. Augmentation was especially important for the solid samples, where the MI estimate was 25% smaller without augmentation. Each such sub-volume was discretized into a binary $n \times n \times n$ voxel lattice, marking cells 1 if they contain a particle and 0 otherwise. The resolution was chosen to be fine enough such that a voxel never contains more than one particle. Samples of smaller volumes, X_1, X_2, etc., were obtained by cropping samples of X_0.

The function $\mathcal{T}$ was parameterized with a 3D convolutional network composed of four successive convolutional blocks with LeakyReLU activation, 0.15 convolutional dropout and adaptive max pooling to target output sizes (20, 10, 5, 2). Starting with $c = 22$ channels, the width is scaled by a 2.5 factor at each block. Finally three fully connected layers with 0.3 dropout[47] were used. Weights were initialized with Xavier uniform that samples weights from a uniform distribution with variance chosen to stabilize gradient magnitudes across layers[48]. For optimization, we used the adaptive first-order stochastic gradient optimizer Adam that uses running estimates of the first and second moments of the gradient (learning rate 3×10^{-5}, batch size 1200) for 1.5×10^5 batches[49]. An exponential moving average estimator was maintained with rate 2.5×10^{-7} and starting value 10^{-5}.

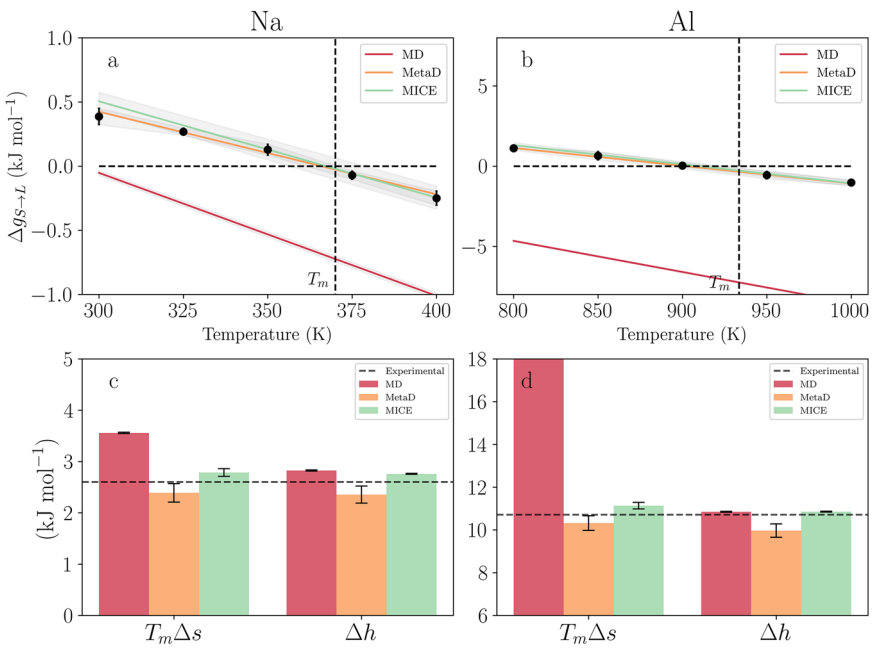

Fig. 3 | Gibbs free energy of melting and its enthalpic-entropic contributions for Na and Al. **a**, **b** Gibbs free energy of melting vs temperature. The data for classical MD and MICE are generated from sampled Δh and the entropy estimates $\Delta s_{\mathrm{id}} + \Delta s_2$ and Δs_{MICE}, respectively. WT-MetaD data is shown by the black points, and the linear fit through them produces estimates of Δh and Δs. **c**, **d** Δh and $T_m \Delta s$ calculated from classical MD, WT-MetaD and MICE (enthalpy estimations for MICE and MD are identical by construction).

We note our experiments showed that, at least for smaller subsystems, equivalent performance in MI estimation can be obtained with smaller networks. However, since the goal of this work is just to demonstrate the applicability of the method, we wanted to avoid any possible effects of hyperparameter tuning, and chose to use the same architecture for all subsystem sizes.

Data Availability

The datasets generated and analyzed during the current study are available from the corresponding authors upon reasonable request. The minimal dataset required to interpret, replicate, and build upon the findings of this study is publicly available at GitHub[50].

Code Availability

The Python code associated with the generation, processing, and analysis of the data presented in this manuscript is available on GitHub[50].

Received: 22 December 2025; Accepted: 31 March 2026;
Published online: 13 April 2026

References

1. Frenkel, D. & Smit, B. *Understanding molecular simulation: from algorithms to applications* (Elsevier, 2023).
2. Barducci, A., Bussi, G. & Parrinello, M. Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Phys. Rev. Lett.* **100**, 020603 (2008).
3. Barducci, A., Bonomi, M. & Parrinello, M. Metadynamics. *Wiley Interdiscip. Rev.: Computational Mol. Sci.* **1**, 826–843 (2011).
4. Valsson, O., Tiwary, P. & Parrinello, M. Enhancing important fluctuations: Rare events and metadynamics from a conceptual viewpoint. *Annu. Rev. Phys. Chem.* **67**, 159–184 (2016).
5. Sutto, L., Marsili, S. & Gervasio, F. L. New advances in metadynamics. *Wiley Interdiscip. Rev.: Computational Mol. Sci.* **2**, 771–779 (2012).
6. Bussi, G. & Laio, A. Using metadynamics to explore complex free-energy landscapes. *Nat. Rev. Phys.* **2**, 200–212 (2020).
7. Kästner, J. Umbrella sampling. *Wiley Interdiscip. Rev.: Computational Mol. Sci.* **1**, 932–942 (2011).
8. Torrie, G. M. & Valleau, J. P. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Computational Phys.* **23**, 187–199 (1977).
9. Miao, Y., Feher, V. A. & McCammon, J. A. Gaussian accelerated molecular dynamics: unconstrained enhanced sampling and free energy calculation. *J. Chem. Theory Comput.* **11**, 3584–3595 (2015).
10. Invernizzi, M. & Parrinello, M. Rethinking metadynamics: from bias potentials to probability distributions. *J. Phys. Chem. Lett.* **11**, 2731–2736 (2020).
11. Invernizzi, M., Piaggi, P. M. & Parrinello, M. Unified approach to enhanced sampling. *Phys. Rev. X* **10**, 041034 (2020).
12. Invernizzi, M. & Parrinello, M. Exploration vs convergence speed in adaptive-bias enhanced sampling. *J. Chem. Theory Comput.* **18**, 3988–3996 (2022).
13. Invernizzi, M. OPES: On-the-fly probability enhanced sampling method. *Nuovo Cimento della Societa Italiana di Fisica C***44** (2021). 2101.06991.
14. Rogal, J., Schneider, E. & Tuckerman, M. E. Neural-network-based path collective variables for enhanced sampling of phase transformations. *Phys. Rev. Lett.* **123**, 245701 (2019).
15. Callen, H. B. Thermodynamics and an introduction to thermostatistics. *John Wiley & Sons* **2** (1980).
16. Lin, S.-T., Maiti, P. K. & Goddard III, W. A. Two-phase thermodynamic model for efficient and accurate absolute entropy of water from molecular dynamics simulations. *J. Phys. Chem. B* **114**, 8191–8198 (2010).
17. Desjarlais, M. P. First-principles calculation of entropy for liquid metals. *Phys. Rev. E* **88**, 062145 (2013).
18. Avinery, R., Kornreich, M. & Beck, R. Universal and accessible entropy estimation using a compression algorithm. *Phys. Rev. Lett.* **123**, 178102 (2019).
19. Martiniani, S., Chaikin, P. M. & Levine, D. Quantifying hidden order out of equilibrium. *Phys. Rev. X* **9**, 011031 (2019).
20. Liu, T. & Simine, L. Deltagzip: Computing biopolymer–ligand binding affinity via Kolmogorov complexity and lossless compression. *J. Chem. Inf. Modeling* **64**, 5617–5623 (2024).
21. Zu, M., Bupathy, A., Frenkel, D. & Sastry, S. Information density, structure and entropy in equilibrium and non-equilibrium systems. *J. Stat. Mech.: Theory Exp.* **2020**, 023204 (2020).
22. Sorkin, B., Be'er, A., Diamant, H. & Ariel, G. Detecting and characterizing phase transitions in active matter using entropy. *Soft Matter* **19**, 5118–5126 (2023).
23. Sorkin, B., Ricouvier, J., Diamant, H. & Ariel, G. Resolving entropy contributions in nonequilibrium transitions. *Phys. Rev. E* **107**, 014138 (2023).
24. Gelman, S. D. & Cohen, G. Nonequilibrium entropy from density estimation. *arXiv preprint arXiv:2405.04877* (2024).
25. Noé, F., Olsson, S., Köhler, J. & Wu, H. Boltzmann generators: Sampling equilibrium states of many-body systems with deep learning. *Science* **365**, eaaw1147 (2019).
26. Schebek, M., Invernizzi, M., Noé, F. & Rogal, J. Efficient mapping of phase diagrams with conditional Boltzmann generators. *Mach. Learn.: Sci. Technol.* **5**, 045045 (2024).
27. Schebek, M., Noé, F. & Rogal, J. Scalable Boltzmann generators for equilibrium sampling of large-scale materials. *arXiv preprint arXiv:2509.25486* (2025).
28. Petersen, M., Roig, G. & Covino, R. Dynamicsdiffusion: Generating and rare event sampling of molecular dynamic trajectories using diffusion models. In *NeurIPS 2023 AI for Science Workshop* (2023). https://openreview.net/forum?id=pwYCCq4xAf.
29. Klein, & Noé, F. Transferable boltzmann generators. *Adv. Neural Inf. Process. Syst*. **37**, 45281 (2024).
30. Nir, A., Sela, E., Beck, R. & Bar-Sinai, Y. Machine-learning iterative calculation of entropy for physical systems. *Proc. Natl. Acad. Sci.* **117**, 30234–30240 (2020).
31. Belghazi, M. I. et al. Mutual information neural estimation. In *International conference on machine learning*, 531–540 (PMLR, 2018).
32. Donsker, M. D. & Varadhan, S. S. Asymptotic evaluation of certain Markov process expectations for large time, I. *Commun. pure Appl. Math.* **28**, 1–47 (1975).
33. Choi, K. & Lee, S. Combating the instability of mutual information-based losses via regularization. In *Uncertainty in Artificial Intelligence*, 411–421 (PMLR, 2022).
34. Wolf, M. M., Verstraete, F., Hastings, M. B. & Cirac, J. I. Area laws in quantum systems: mutual information and correlations. *Phys. Rev. Lett.* **100**, 070502 (2008).
35. Piaggi, P. M., Valsson, O. & Parrinello, M. Enhancing entropy and enthalpy fluctuations to drive crystallization in atomistic simulations. *Phys. Rev. Lett.* **119**, 015701 (2017).
36. Batzner, S. et al. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.* **13**, 2453 (2022).
37. Satorras, V. G., Hoogeboom, E. & Welling, M. E (n) equivariant graph neural networks. In *International conference on machine learning*, 9323–9332 (PMLR, 2021).
38. Gasteiger, J, Groß, J. & Günnemann, S. In *International Conference on Learning Representations* (2020).
39. Thompson, A. P. et al. LAMMPS-a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Computer Phys. Commun.* **271**, 108171 (2022).
40. Bonomi, M. et al. Plumed: A portable plugin for free-energy calculations with molecular dynamics. *Computer Phys. Commun.* **180**, 1961–1972 (2009).

41. Promoting transparency and reproducibility in enhanced molecular simulations. *Nat. Methods* **16**, 670–673 (2019).
42. Tribello, G. A., Bonomi, M., Branduardi, D., Camilloni, C. & Bussi, G. Plumed 2: New feathers for an old bird. *Computer Phys. Commun.* **185**, 604–613 (2014).
43. Wilson, S. R., Gunawardana, K. G. S. H. & Mendelev, M. I. *J Chem. Phys.* **142**, 134705 (2015).
44. Mendelev, M., Kramer, M., Becker, C. A. & Asta, M. Analysis of semi-empirical interatomic potentials appropriate for simulation of crystalline and liquid Al and Cu. *Philos. Mag.* **88**, 1723–1750 (2008).
45. Bussi, G., Donadio, D. & Parrinello, M. *J chem. phys.* **126**, https://doi.org/10.1063/1.2408420 (2007).
46. Parrinello, M. & Rahman, A. Polymorphic transitions in single crystals: A new molecular dynamics method. *J. Appl. Phys.* **52**, 7182–7190 (1981).
47. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I. & Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **15**, 1929–1958 (2014).
48. Glorot, X. & Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 249–256 (JMLR Workshop and Conference Proceedings, 2010).
49. Kingma, D. P. & Ba, J. In *International Conference on Learning Representations (ICLR)* (2015).
50. Ben Shimon, Y., Bar-Sinai, Y. & Hirshberg, B. mice-free-energy. https://github.com/ybs-lab/mice-free-energy (2025). GitHub repository.

Acknowledgements

We thank Pablo Piaggi for sharing his code and useful discussions.

Author contributions

Barak H. and Yohai B.S. designed the research, wrote the manuscript, and analyzed the results. Yamin B.S. performed all the simulations, led the analysis, and wrote the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41524-026-02076-z.

Correspondence and requests for materials should be addressed to Barak Hirshberg or Yohai Bar-Sinai.