Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2026, **5**, 415

Understanding and mitigating distribution shifts for universal machine learning interatomic potentials

Tobias Kreiman *[a] and Aditi S. Krishnapriyan[ab]

Machine Learning Interatomic Potentials (MLIPs) are a promising alternative to expensive *ab initio* quantum mechanical molecular simulations. Given the diversity of chemical spaces that are of interest and the cost of generating new data, it is important to understand how universal MLIPs generalize beyond their training distributions. In order to characterize and better understand distribution shifts in MLIPs—that is, changes between the training and testing distributions—we conduct diagnostic experiments on chemical datasets, revealing common shifts that pose significant challenges, even for large universal models trained on extensive data. Based on these observations, we hypothesize that current supervised training methods inadequately regularize MLIPs, resulting in overfitting and learning poor representations of out-of-distribution systems. We then propose two new methods as initial steps for mitigating distribution shifts for MLIPs. Our methods focus on test-time refinement strategies that incur minimal computational cost and do not use expensive *ab initio* reference labels. The first strategy, based on spectral graph theory, modifies the edges of test graphs to align with graph structures seen during training. Our second strategy improves representations for out-of-distribution systems at test-time by taking gradient steps using an auxiliary objective, such as a cheap physical prior. Our test-time refinement strategies significantly reduce errors on out-of-distribution systems, suggesting that MLIPs are capable of and can move towards modeling diverse chemical spaces, but are not being effectively trained to do so. Our experiments establish clear benchmarks for evaluating the generalization capabilities of the next generation of MLIPs. Our code is available at https://tkreiman.github.io/projects/mlff_distribution_shifts/.

Received 11th June 2025
Accepted 17th November 2025

DOI: 10.1039/d5dd00260e

rsc.li/digitaldiscovery

1 Introduction

Understanding the quantum mechanical properties of atomistic systems is crucial for the discovery and development of new molecules and materials. Computational methods like Density Functional Theory (DFT) are essential for studying these systems, but the high computational demands of such methods limit their scalability. Machine Learning Interatomic Potentials (MLIPs) have emerged as a promising alternative, learning to predict energies and forces from reference quantum mechanical calculations. MLIPs are faster than traditional *ab initio* methods, and their accuracy is rapidly improving for modeling complex atomistic systems.[5,7,29,57]

Given the computational expense of *ab initio* simulations for all chemical spaces of interest, there has been a push to train larger and more accurate MLIPs, designed to work well across many different systems. Developing models with general representations that accurately capture diverse chemistries has the potential to reduce or even eliminate the need to recollect data and retrain a model for each new system. To determine which systems an MLIP can accurately describe and to assess the reliability of its predictions, it is important to understand how MLIPs generalize beyond their training distributions. This understanding is essential for applying MLIPLs to new and diverse chemical spaces, ensuring that they perform well not only on the data they were trained on, but also on unseen, potentially more complex systems.

In other fields of machine learning (ML), model generalization has been extensively studied through the lens of distribution shifts: changes between the training and testing distributions.[43,61,63,70,71] In computer vision, for example, a distribution shift would occur if a model trained on color images is evaluated on black and white images. The extensive work to both categorize and mitigate distribution shifts in the broader field of ML has helped practitioners determine where models can be reliably applied.

We conduct an in-depth exploration to identify and understand distribution shifts for MLIPs. On example chemical datasets, we find that many large-scale models struggle with common distribution shifts[6,46,50,58] (see Section 3). These generalization challenges suggest that current supervised training methods for MLIPs overfit to training distributions and do not enable MLIPs to generalize accurately. We demonstrate that there are multiple reasons that this is the case, including challenges associated with poorly-connected graphs and

[a]*UC Berkeley, USA. E-mail: tkreiman@berkeley.edu*
[b]*LBNL, USA*

learning unregularized representations, evidenced by jagged predicted potential energy surfaces for out-of-distribution systems.

Building on our observations, we take initial steps to mitigate distribution shifts for MLIPs without test set reference labels by proposing two approaches: test-time radius refinement and test-time training.[27,42,62] For test-time radius refinement, we modify the construction of test-graphs to match the training Laplacian spectrum, overcoming differences between training and testing graph structures. For test-time training (TTT), we address distribution shifts by taking gradient steps on an auxiliary objective at test time. Analogous to self-supervised objectives in computer vision TTT works,[27,36,62] we use an efficient prior as a target to improve representations at test time.

Although completely closing the out-of-distribution to in-distribution gap remains a challenging open machine learning problem,[27,62] our extensive experiments show that our test-time refinement strategies are effective in mitigating distribution shifts for MLIPs. Our experiments demonstrate that low quality data can be used to improve generalization for MLIPs, and they establish clear benchmarks that highlight ambitious but important generalization goals for the next generation of MLIPs.

We summarize our main contributions here:

(1) We run diagnostic experiments on different chemical datasets to characterize and understand common distribution shifts for MLIPs in Section 3.

(2) Based on (1), we take first steps at mitigating MLIP distribution shifts in Section 4 with two test-time refinement strategies.

(3) The success of these methods, validated through extensive experiments in Section 5, suggests that MLIPs are not being adequately trained to generalize, despite current models having the expressivity to close the gap on the distribution shifts explored in Section 3.

2 Related work

2.1 Distribution shifts

Since most machine learning algorithms assume that the training and testing data are independently and identically distributed (the I.I.D. assumption), a long line of work has studied violations of this assumption, commonly referred to as distribution shifts.[17,40,51,54] Sugiyama *et al.*[61] demonstrated how to perform importance weighted cross validation to perform model selection under distribution shifts. Methods have been proposed to measure and improve the robustness of models to distribution shifts in images[63,71] and language.[70] Numerous methods have been proposed to tackle distribution shifts including, but not limited to, techniques based on meta learning[43] and ensembles.[72]

Recent work has also begun identifying generalization challenges with MLIPs.[10,49] Deng *et al.*[18] find that MLIPs systematically underpredict energy surfaces, and that this underprediction can be ameliorated with a small number of fine-tuning steps on reference calculations. Our experiments complement these initial findings of underestimation, and we also identify other types of distribution shifts, like connectivity and atomic feature shifts. Our proposed test-time refinement solutions are also able to mitigate distribution shifts without any reference data, and they provide insights into why MLIPs are unable to generalize.

2.2 Multi-fidelity machine learning interatomic potentials

Behler & Parrinello[9] popularized the use of machine learning for modeling potentials, leading to numerous downstream applications[2] and refinements to model increasingly complicated systems.[19] More recent work has explored training MLIPs with observables and unsupervised objectives,[22,26,35,52,68] distilling MLIPs with physical constraints,[1] and using multiple levels of theory during training. Amin *et al.*[1] found that knowledge distillation can enable smaller models to outperform larger models in certain specialized tasks, suggesting that the larger MLIPs may not have been trained in a way that fully leverages their capacity. Jha *et al.*,[44] Gardner *et al.*,[28] and Shui *et al.*[59] leveraged cheap or synthetic data to improve data efficiency and accuracy. Ramakrishnan *et al.*[53] popularized the Δ-learning approach,[11] where a model learns to predict the difference between some prior and the reference quantum mechanical targets. Multi-fidelity learning generalizes Δ-learning by building a hierarchy of models that predict increasingly accurate levels of theory.[23,33,37,66] Making predictions in the hierarchical multi-fidelity setting corresponds to evaluating a baseline fidelity level and then refining this prediction with models that provide corrections to more accurate levels of theory in the hierarchy.

Our work differs from these works in several ways. We focus on developing training strategies that address distribution shifts. In contrast to prior multi-fidelity works, we learn representations from multiple levels of theory using pre-training, fine-tuning, and joint-training objectives. Rather than fine-tuning all the model weights like in Jha *et al.*,[44] Gardner *et al.*,[28] and Shui *et al.*,[59] we explore freezing and regularization techniques that enable test-time training. Our new test-time objectives update the model's representations when faced with out-of-distribution examples, improving performance on out-of-distribution systems. Multi-fidelity approaches by themselves do not tackle the challenge of transferring to new, unseen systems at test-time. Nevertheless, combining our training strategies with other multi-fidelity approaches presents an interesting direction for future work.

2.3 Test-time training

The test-time training (TTT) framework adapts predictive models to new test distributions by updating the model at test-time with a self-supervised objective Sun *et al.*[62] Sun *et al.*[62] demonstrated that forcing a model to use features learnt from a self-supervised objective during the main task allows the model to adapt to out-of-distribution examples by tuning the self-supervised objective. Follow up work showed the benefits of TTT across computer vision and natural language processing, exploring a range of self-supervised objectives.[27,36,42]

3 Distribution shifts for machine learning interatomic potentials

3.1 Problem setup and background

MLIPs approximate molecule-level energies and atom-wise forces for a chemical structure by learning neural network parameters from data. For a given a molecular structure, the input to the ML model consists of two vectors: $\mathbf{r} \in \mathbb{R}^{n\times 3}$, $\mathbf{z} \in \mathbb{R}^{n\times d}$, where n represents the number of atoms in the molecule, $\mathbf{r}$ are the atomic positions, and $\mathbf{z}$ are the features of the atom, such as atomic numbers or whether an atom is fixed or not. The model outputs $\hat{E} \in \mathbb{R}$, $\hat{\mathbf{F}} \in \mathbb{R}^{n\times 3}$, which are the predicted total potential energy of the molecule and the predicted forces acting on each atom. The learning objective is typically formulated as a supervised loss function, which measures the discrepancy between the predicted energies and forces and reference energies and forces:

$$\mathcal{L}(\mathbf{F}, E) = \lambda_{\mathrm{E}} \|E_{\mathrm{ref}} - \hat{E}\|_2^{\,2} + \lambda_{\mathrm{F}} \sum_{i=1}^{n} \|\mathbf{F}_{i,\mathrm{ref}} - \hat{\mathbf{F}}\|_2^{\,2}, \quad (1)$$

where λ_{E}, λ_{F} are hyperparameters.

Most modern MLIPs are implemented as graph neural networks (GNNs).[31] Consequently, $\hat{E}$ and $\hat{\mathbf{F}}$ are functions of $\mathbf{z}$, $\mathbf{r}$, and $A \in \mathbb{R}^{n\times n}$, the adjacency matrix representing the molecule:

$$\hat{E}, \hat{\mathbf{F}} = f(\mathbf{z}, \mathbf{r}, \mathrm{A}) \quad (2)$$

The atoms in the molecule are modeled as nodes in a graph, and edges are specified by the adjacency matrix that includes connections to all atoms within a specified radius cutoff.[5,29] The adjacency matrix fully determines a graph structure, and thus defines the graph over which the GNN performs its computation.

3.2 Criteria for identifying distribution shifts

In this section, we formalize criteria for identifying distribution shifts—that is, changes from the training to the testing distribution—based on the features, labels, and graph structures in chemical datasets. The same way that the computer vision community identifies distribution shifts due to differences in image resolution, color profile, and,[27,38,54,62] we seek to define distribution shifts for MLIPs that broadly encompass the diversity of chemical spaces. We also note that distribution shifts can occur independently along each dimension: *e.g.*, a shift in features does not necessarily imply a shift in labels (see Section A.5 for details). This categorization provides a framework for understanding the types of distribution shifts anMLIP may encounter (see Fig. 1). This understanding motivates the refinement strategies described in Section 4 that take first steps at mitigating these shifts, providing insights into why MLIPs are susceptible to these shifts in the first place.

3.2.1 Distribution shifts in atomic features (z). Distribution shifts in atomic features refer to any change in the atomic composition of a chemical system. This includes, but is not limited to, cases where models are trained on systems containing mixtures of organic elements but tested on structures composed solely of carbon, or cases where there is a shift in system size between training and testing. Fig. 1 illustrates an atomic feature distribution shift by comparing a small carbon dioxide molecule to larger molecular system containing 91 atoms.

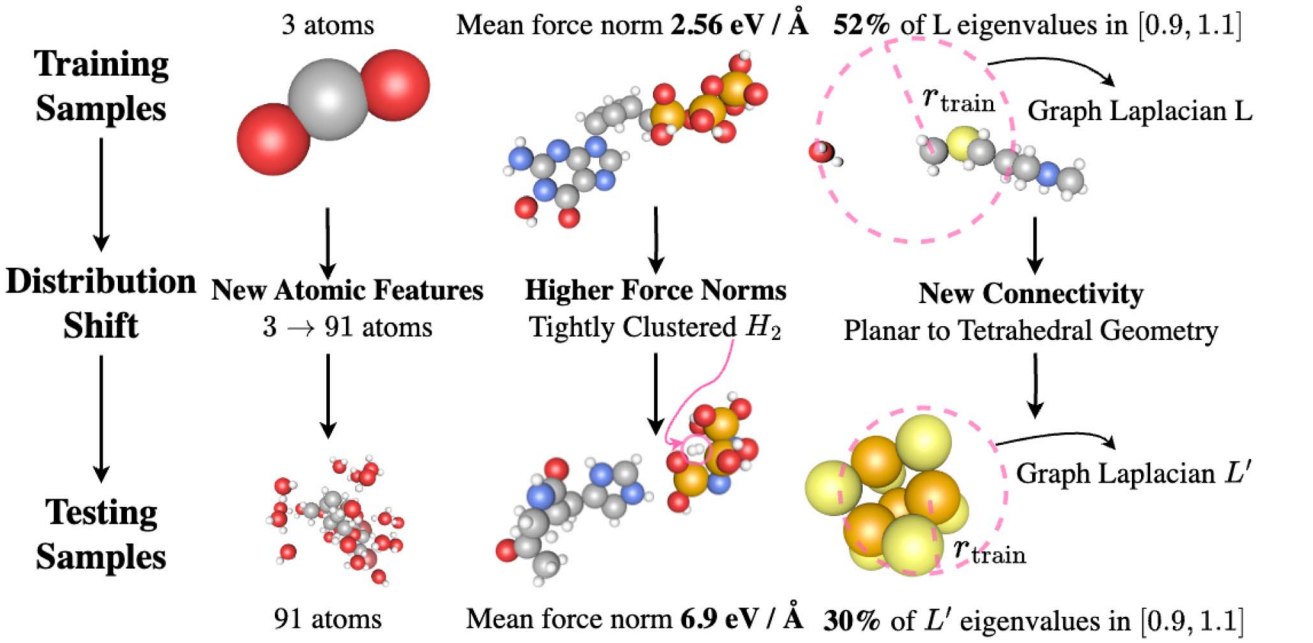


Fig. 1 Distribution shifts for MLIPs. We visualize distribution shifts based on changes in features, labels, and graph structure. Typical training samples from SPICE[20] and new systems from SPICEv2 (ref. 21) are displayed. An atomic feature shift is illustrated by comparing a three-atom molecule with a larger molecular system containing 91 atoms (left). A force norm shift is shown by the close proximity of an H_2 molecule (circled in pink), leading to high force norms (middle). A connectivity shift is shown by the tetrahedral geometry in P_4S_6, which differs from the typical planar geometry seen during training (right).

3.2.2 Distribution shifts in forces (F). An MLIP may also encounter a distribution shift in the force labels it predicts. A model trained on structures close to equilibrium, with low force magnitudes, might be tested on a structure with higher force norms. Fig. 1 shows an example of a tightly clustered H_2 molecule, which leads to a force norm distribution shift.

3.2.3 Distribution shifts in graph structure and connectivity (A). Since many MLIPs are implemented as GNNs, they may encounter distribution shifts in the graph structure defined by A. We refer to these as connectivity distribution shifts because A determines the graph connectivity used by the GNN. Connectivity distribution shifts are particularly common in molecular datasets, where one could encounter a benzene ring at test time, despite only having trained on long acyclic structures. Fig. 1 provides an example of a connectivity distribution shift, going from planar training structures to a tetrahedral geometry at test time.

We identify connectivity distribution shifts by analyzing the eigenvalue spectra of the normalized graph Laplacian:

$$L = I - (D)^{-\frac{1}{2}} A (D)^{-\frac{1}{2}}, \tag{3}$$

where $D \in \mathbb{R}^{n \times n}$ is the degree matrix ($D_{ii} = \text{degree}(\text{node}_i)$ and $D_{ij} = 0$ for $i \neq j$, $A_{ij} = 1$ if $\|r_i - r_j\|_2 \leq r_{\text{cutoff}}$ and 0 otherwise), and I is the identity. L has eigenvalues $\lambda_0, \leq \lambda_1, \leq \ldots \leq \lambda_{n-1}$, where $\lambda_i \in [0, 2] \forall i$, and the multiplicity of the 0 eigenvalue equals the number of connected components in the graph.

Following previous work,[16,67] we can compare structural differences between graphs by using the spectral distance.[45] Since Laplacian spectra are theoretically linked to information propagation in GNNs,[32,67] the spectral distance is a natural choice for comparing molecular graphs (see Sections 4.1 and A.2 for more details).

3.2.4 Physical origins of distribution shifts. An MLIP might encounter a distribution shift along these three axes (atomic features, force norms, and connectivity) due to a number of physical phenomena. For example, high force norms might be encountered by applying an MLIP trained on near-equilibrium structures to analyze transition state regions or when running MD simulations at a high temperature. Nevertheless, we explicitly define these distribution shifts along three abstract axes in order to encompass the broad types of chemistries an MLIP might encounter.

We note that other works in ML also define "abstract" types of distribution shifts that are independent of their underlying causes.[27,54,62,70] For instance, a distribution shift in the color of images could result from changes in lighting or from using a new camera. Both nevertheless constitute a shift in image color. Categorizing shifts in this abstract way makes it possible to diagnose where the shift occurs (*e.g.*, in color) and to develop general methods to mitigate them. Similarly, by treating force norms as a type of distribution shift, we believe that general methods for handling such shifts will enable MLIPs to better model transition regions, high-temperature dynamics, and beyond.

3.2.5 Observed distribution shifts for large models. We contextualize the aforementioned distribution shifts by considering seven large models: eSEN (on OMol and OMat24), MACE (trained on SPICE and OMat24), EquiformerV2 (trained on OC20 and OMat24), and JMP.[4,6,46,48,50,58] MACE-OFF is a biomolecules universal model trained on 951k structures primarily from the SPICE dataset.[20] We examine MACE, eSEN, and EquiformerV2 models trained on 100M+ structures from OMat24.[4] We also evaluate an EquiformerV2 model trained on 100M+ structures from OC20 (ref. 12) and the eSEN-md model trained on 100M+ structures from OMol.[25,48] The JMP model is trained on 100M+ structures from OC20, OC22, ANI-1x, and Transition-1x.[12,55,60,64] These models represent seven of the largest open-source MLIPs to date, and they have been trained on some of the most extensive datasets available. We focus on these models since their scale is designed for tackling broad chemical spaces.

We examine the generalization ability of MACE-OFF by testing it on 10k new molecules from the SPICEv2 dataset[21] not included in the MACE-OFF training set. A molecule is defined as out-of-distribution if it is more than 1 standard deviation away from the mean training data force norm, system size, or connectivity (with respect to the spectral distance defined above Section 3.2). Despite its scale, MACE-OFF performs worse by an order of magnitude on out-of-distribution systems (see Fig. 2c). We also evaluate JMP on the ANI-1x[60] test set defined in Shoghi *et al.*[58] and eSEN on the OMol validation set.[48] While JMP and eSEN have lower absolute errors than MACE-OFF since they were trained on significantly more data, they still suffer predictably from force norm, connectivity, and atomic feature distribution shifts (see Fig. 2f and g). We also find significant degradation in performance at high charge and spin for eSEN (see Fig. 21), providing further evidence that distribution shifts still pose challenges, even for a new model trained on one of the largest currently available datasets.

We focus on force norm distribution shifts for the models trained on OMat24 and OC20, since connectivity is more uniform across bulk materials and catalysts, where atoms are packed tightly into a periodic cell. For the MACE, EquiformerV2, and eSEN models trained on OMat24, we evaluate performance on the OMat24 validation set.[4] These models still clearly perform worse as force norms deviate from the majority of the training distribution (see Fig. 2a, c and d). EquiformerV2 also struggles with high force norm structures when evaluated on the validation out-of-distribution set from OC20 (ref. 12) (see Fig. 2b).

3.2.6 Observations. Training larger models with more data is one approach to address these distribution shifts (for example, with active learning[47,65]). However, doing so can be computationally expensive. Our diagnostic experiments also indicate that scale alone might not fully address distribution shifts, as naively adding more in-distribution data does not necessarily help large models generalize better (see Fig. 2). The diversity of chemical spaces makes it exceedingly difficult to know the exact systems that an MLIP will be tested on *a priori*, making it challenging to curate the perfect training set. These observations lead us to develop strategies that mitigate distribution shifts by modifying the training and testing procedure of MLIPs. Importantly, these refinement strategies can be combined with any further architecture and data advances.

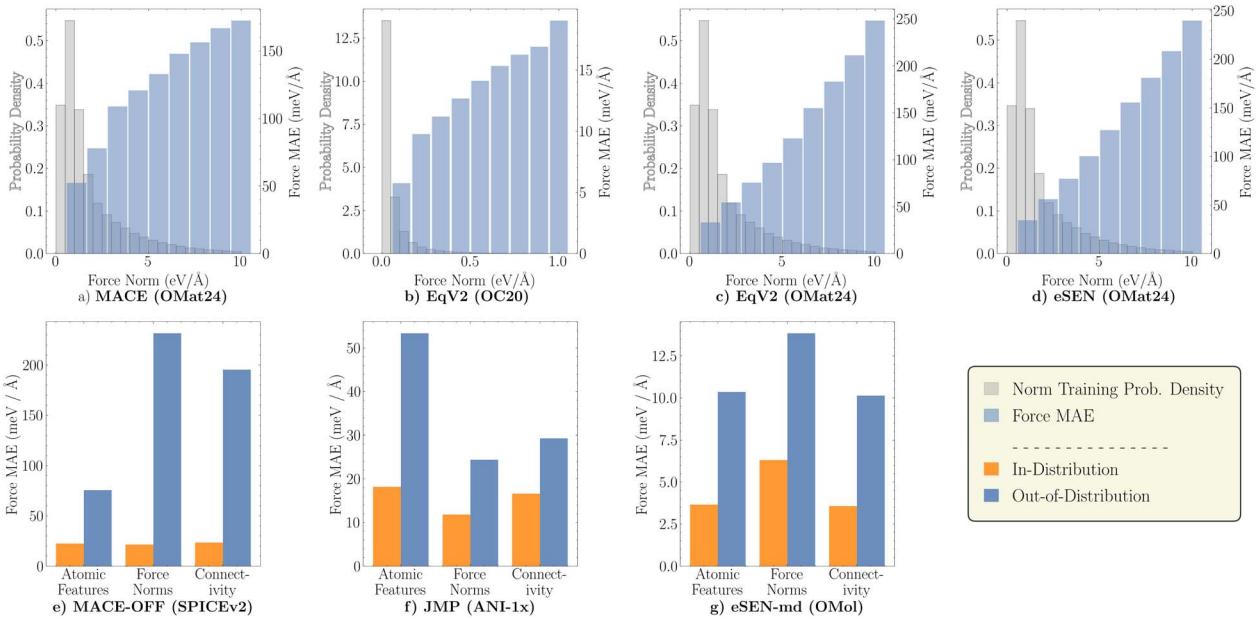


Fig. 2 Distribution shifts for large models. We study distribution shifts on seven of the largest open-source MLIPs designed for broad chemical spaces. We evaluate MACE (a), EquiformerV2 (c), and eSEN (d) on the OMat24 dataset. We evaluate EquiformerV2 (b) on the OC20 out-of-distribution validation set. We evaluate MACE-OFF (e) on 10k new molecules from SPICEv2. We evaluate JMP (f) on the ANI-1x test set. We evaluate eSEN-md (g) on the OMol dataset. A molecule is considered out-of-distribution if it is more than 1 standard deviation away from the mean training force norm, system size, or connectivity (with respect to the spectral distance defined above Section 3.2). Despite their scale, these large universal models have 2–10× larger force mean absolute errors (MAE) when encountering distribution shifts.

4 Mitigating distribution shifts with test-time refinement strategies for machine learning interatomic potentials

Based on the generalization challenges for universal models (see Section 3), we hypothesize that many MLIPs are severely overfitting to the training data, resulting in a failure to learn generalizable representations. Building on our observations in Section 3 and to test this hypothesis, we develop two test-time refinement strategies that also mitigate distribution shifts. We focus on test time evaluations, *i.e.*, with access to test molecular structures but without access to reference labels. First, by studying the graph Laplacian spectrum, we investigate how MLIPs, and GNNs in general,[8] tend to overfit to the regular and well-connected training graphs. In Section 4.1, we address connectivity distribution shifts by aligning the Laplacian eigenvalues of a test structure with the connectivities of the training distribution. Second, we show that MLIPs are inadequately regularized, resulting in poor representations of out-of-distribution systems. We incorporate inductive biases from a cheap physical prior using our pre-training and test-time training procedure (Section 4.2) to regularize the model and learn more general representations, evidenced by smoother predicted potential energy surfaces. The effectiveness of these test-time refinement strategies, validated through extensive experiments in Sections 5 and A.3, may indicate that MLIPs are currently poorly regularized and overfit to graph structures seen during training, hindering broader generalization.

4.1 Test-time radius refinement

We hypothesize that MLIPs tend to overfit to the specific graph structures encountered during training. We can characterize graph structures by studying the Laplacian spectrum of a graph. At test time, we can then identify when an MLIP encounters a graph with a Laplacian eigenvalue distribution that significantly differs from the training graphs (see 3.2). To address this shift, we propose updating the test graph to more closely resemble the training graphs, thereby mitigating connectivity distribution shifts. Since the adjacency matrix A and graph Laplacian L are typically generated by a radius graph, we refine the radius cutoff at test time. Instead of using a fixed radius cutoff r_{train} for both training and testing, adjusting the radius cutoff at test time can help achieve a connectivity that more closely resembles the training graphs.

Formally, for each test structure j, we search over k new radius cutoffs $[r_i]_{i=1}{}^{k}$, calculate the new eigenvalue spectra for $L^{(j)}$ induced by the new cutoff r_i, and select the r_i that minimizes the difference between the eigenvalue spectra of the new graph and the training graphs (see Fig. 3):

$$r_{\text{test}}^{(j)} = \arg\min_{[r_i]_{i=1}{}^{k}} D\left(\lambda_{\text{train}}, \lambda\left(L^{(j)}(r_i)\right)\right), \tag{4}$$

where λ_{train} is the training distribution of eigenvalues, $\lambda(L^{(j)}(r_i))$ is the Laplacian spectrum for sample j generated with radius cutoff r_i, and D is some distance function. We choose the squared spectral distance:

$$D\left(\lambda_{\text{train}}, \lambda\left(L^{(j)}(r_i)\right)\right) = \sum_l \left(\bar{\lambda}_l - \lambda\left(L^{(j)}(r_i)\right)_l\right)^2, \tag{5}$$

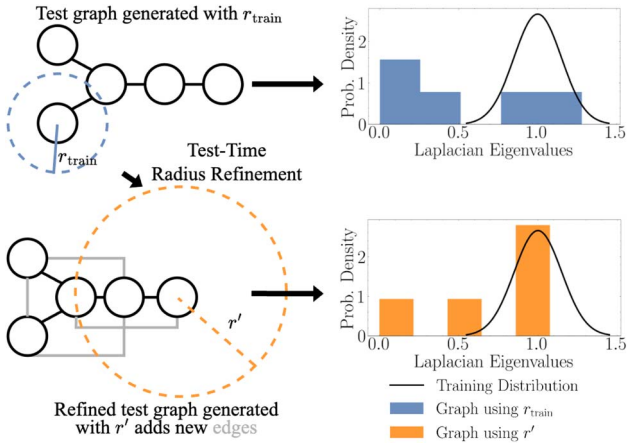


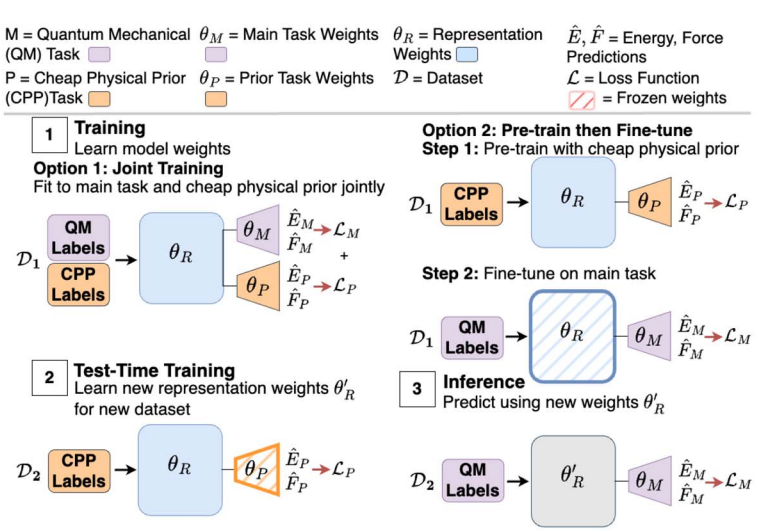
Fig. 3 Test-time radius refinement. MLIPs tend to overfit to the well-connected graphs seen during training, which can be identified by the clustering of Laplacian eigenvalues around 1. To mitigate connectivity distribution shifts at test time, we find the optimal radius cutoff, which aligns the Laplacian eigenvalues of test graphs with those of the training distribution.

where, following previous work, $\bar{\lambda}$ is the average Laplacian spectrum of the training distribution with spectra padded with zeros to accommodate different sized graphs, and l indexes each individual eigenvalue.[16,45] In other words, for each test structure indexed by j, we search over k different trial radii and select the one that yields a Laplacian spectra most similar to the training distribution. While averaging the spectra across the training distribution provides a lossy representation of the training connectivities, it is computationally impractical to compare each new test structure to all training graphs individually. One alternative is to count the number of training graphs within a certain cutoff of the spectral distance to assess how far a test graph is from the training distribution. However, this measure is highly correlated with the simpler spectral distance metric, eqn (5) (see Fig. 19). Consequently, while per-sample comparisons could be useful in some cases, we use the more computationally efficient spectral distance metric, eqn (5), in our experiments. For further details and theoretical motivation, see Sections A.1 and A.2.

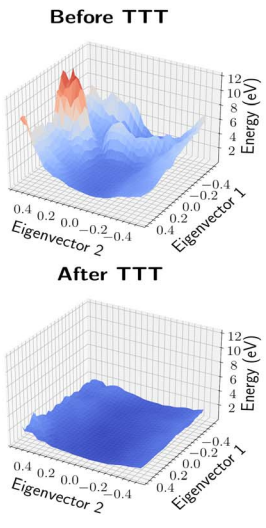
Our experiments show that this procedure virtually never deteriorates performance, as one can always revert to the same radius cutoff used during training (see Section 5). Additionally, we emphasize that we update the envelope function with the new radius to ensure smoothness of the potential energy surface. We verify that this does maintain energy conservation in Table 3. This refinement method addresses the source of connectivity distribution shifts and serves as an efficient and effective strategy for handling new connectivities.

4.2 Test-time training using cheap priors

We further hypothesize that the current supervised training procedure for MLIPs can lead to overfitting, leading to poor representations for out-of-distribution systems and jagged potential energy landscape predictions (see Fig. 4b for an example on salicylic acid). To address this, we propose introducing inductive biases through improved training and inference strategies to smooth the predicted energy surfaces. The smoother energy landscape from the improved training indicates that the model may have learned more robust



(a) Test-Time Training (TTT)



(b) Predicted Potential Energy Surface

Fig. 4 Test-time training mitigates distribution shifts and smooths predicted potential energy surfaces. We hypothesize that due to overfitting, the predicted potential energy surfaces are jagged for out-of-distribution systems. Our proposed test-time training method (TTT, (a)) regularizes MLIPs by incorporating inductive biases into the model using a cheap prior. Test-time training first learns useful representations from the prior using either joint-training or a pre-train, freeze, and fine-tune approach. TTT then updates the representations at test-time using the prior to improve performance on out-of-distribution samples. We plot the predicted potential energy surface from a GemNet-dT model along the 2 principal components of the Hessian for salicylic acid, a molecule not seen during training, before and after test-time training (b). TTT effectively smooths the potential energy landscape and improves errors.

representations, mitigating force norm, atomic feature, and connectivity distribution shifts.

We represent these inductive biases as cheap priors, such as classical force fields or simple ML models. In our main experiments (see Section 5), we use the sGDML model[14] and the semi-empirical GFN2-xtb,[3] both of which are broadly used across the periodic table and contain a number of physical inductive biases, like spatial symmetries and electrostatic effects. We note that our proposed method can work with any other force field. These priors can evaluate thousands of structures per second using only a CPU, making them computationally efficient for test-time use. First, we describe our pre-training procedure, which ensures the MLIP learns useful representations from the cheap prior. By leveraging these representations, we can smooth the predicted energy landscape and mitigate distribution shifts by taking gradient steps with our test-time training (TTT) procedure.

4.2.1 Pre-training with cheap physical priors. We propose a training strategy that first pre-trains on energy and force targets from a cheap prior and then fine-tunes the model on the ground truth quantum mechanical labels. Our loss function for one structure is defined as:

$$\mathcal{L}(\mathbf{F}^{\mathrm{M}}, E^{\mathrm{M}}, \mathbf{F}^{\mathrm{P}}, E^{\mathrm{P}}) = \mathcal{L}_{\mathrm{M}} + \mathcal{L}_{\mathrm{P}} = \sum_{l \in \{\mathrm{M},\mathrm{P}\}} \left(\lambda_{E^l} \| E^l - \hat{E}^l \|_2^{\,2} + \lambda_{F^l} \sum_{i=1}^{n} \| \mathbf{F}_i^{\,l} - \hat{\mathbf{F}}_i^{\,l} \|_2^{\,2} \right), \quad (6)$$

where $\hat{E}$,$\hat{\mathbf{F}}$ are the predicted energy and forces, and M and P denote the main and prior task, respectively. During pre-training, gradient steps are initially only taken on the prior objective, corresponding to $\mathcal{L}_{\mathrm{P}}$. For fine-tuning, the representation parameters, θ_{R}, learnt from the prior are kept frozen, and the main task parameters, θ_{M}, are updated by training only on the main task loss, $\mathcal{L}_{\mathrm{M}}$. Pre-training and fine-tuning can also be merged and the model can be jointly trained on both the cheap prior targets and the expensive DFT targets (see Fig. 4a). This corresponds to training on $\mathcal{L}_{\mathrm{P}} + \mathcal{L}_{\mathrm{M}}$. Freezing or joint-training both force the main task head to rely on features learnt from the prior. This approach acts as a form of regularization, resulting in more robust representations. It enables the prior to be used to improve the features extracted from an out-of-distribution sample at test time, improving main task performance. For more details on the necessity of proper pre-training for test-time training, see Section A.1.

4.2.2 TTT implementation details. For clarity, let us separate our full model into its three components: $g_{\theta_{\mathrm{R}}}$ (the representation model), $h_{\theta_{\mathrm{M}}}$ (the main task head), and $h_{\theta_{\mathrm{P}}}$ (the prior task head). The representation parameters, θ_{R}, are learned by minimizing $\mathcal{L}$ during joint training (see eqn (6)), or by minimizing $\mathcal{L}_{\mathrm{P}}$ during pre-training and then freezing them during the fine-tuning phase. Test-time training involves the following steps:

(1) Updating representation parameters. At test-time, we update θ_{R} by minimizing the prior loss, $\mathcal{L}_{\mathrm{P}}$, on samples from the test distribution $\mathcal{D}_{\mathrm{test}}$, which are labeled by the cheap prior. This is expressed as:

$$\theta_{\mathrm{R}}^{'} = \arg\min_{\theta_{\mathrm{R}}} \mathbb{E}_{(r,z,\mathbf{F}^{\mathrm{p}},E^{p}) \sim \mathcal{D}_{\mathrm{test}}} \left[\mathscr{L}_{\mathrm{P}} \left(h_{\theta_{\mathrm{P}}} \circ g_{\theta_{\mathrm{R}}}(r,z), \mathbf{F}^{\mathrm{p}}, E^{p} \right) \right]. \quad (7)$$

During this process, the prior head parameters, θ_{P}, are kept frozen during test-time updates. This incorporates inductive biases about the out-of-distribution samples into the model, regularizing the energy landscape and helping the model generalize (see Fig. 4b and 16).

(2) Prediction on test set. Once the representation parameters are updated, we predict the main task labels for the test set using the newly adjusted representation:

$$\hat{E}, \hat{\mathbf{F}} = h_{\theta_{\mathrm{M}}} \circ g_{\theta_{\mathrm{R}}^{'}}(\mathbf{r}, \mathbf{z}). \quad (8)$$

We recalculate the parameters $\theta_{\mathrm{R}}^{'}$ with eqn (7) when a new out-of-distribution region is encountered (*i.e.*, when testing on a new system). See Fig. 4a for an outline of our method.

We formalize the intuition behind TTT for MLIPs in the following theorem, where we look at TTT with a simple Lennard-Jones prior:[56]

Theorem 4.1 *If the reference energy calculations asymptotically go to ∞ as pairwise distances go to 0, then there exist test-time training inputs such that a gradient step on the prior loss, with the Lennard-Jones potential, reduces the main task loss on those inputs.*

We prove Theorem 4.1 by showing that there exist points where the errors on the prior and main task are correlated ($\mathrm{sign}(\hat{E}^{\mathrm{P}} - E^{\mathrm{P}}) = \mathrm{sign}(\hat{E}^{\mathrm{M}} - E^{\mathrm{M}})$), and that the main task head and the prior task head use similar features ($\theta_{\mathrm{P}}{}^{T}\theta_{\mathrm{M}} > 0$). Building off of the theoretical result in Sun *et al.*,[62] this implies that TTT on these points with prior labels improves main task performance. For a detailed proof, see Section A.2.

5 Experiments

We conduct experiments on chemical datasets to both identify the presence of distribution shifts and evaluate the effectiveness of our test-time refinement strategies to mitigate these shifts. In Section 5.1, we find distribution shifts on the SPICE dataset with the MACE-OFF universal model.[20,46] In Section 5.2, we explore extreme distribution shifts and demonstrate that our test-time refinement strategy enables stable simulations on new molecules, even when trained on a limited dataset of 3 molecules from the MD17 dataset.[13] Finally, in Section A.3.4, we assess how our test-time refinement strategy can handle high force norms in the MD22 dataset when the model is trained only on low force norms. Although matching in-distribution performance (without access to ground truth labels) remains a challenging open machine learning problem,[27,62] our experiments indicate that test-time refinement strategies are a promising initial step for addressing distribution shifts with MLIPs. The improvements from these test-time refinement strategies also suggest that MLIPs can be trained to learn more general representations that are resilient to distribution shifts. Additional experiments with more models, datasets, and priors are provided in Section A.3.

5.1 Distribution shifts: training on SPICE and testing on SPICEv2

We investigate distribution shifts from the SPICE dataset to the SPICEv2 dataset[20,21] by analyzing the MACE-OFF universal

model.[46] As shown in Fig. 6, 7, and 11, we observe that despite being trained on 951k data points and scaled to 4.7M parameters, MACE-OFF experiences force norm, connectivity, and atomic feature distribution shifts when evaluated on 10k new molecules from SPICEv2.[21] Any deviation from the training distribution, shown in gray, predictably results in an increase in force error.

We evaluate the effectiveness of our test-time refinement strategies in mitigating these distribution shifts. For the MACE-OFF model, we implement test-time radius refinement (RR) by searching over 10 different radius cutoffs and selecting the one that best matches the training Laplacian eigenvalue distribution (see Section 4.1). We also train a GemNet-T model on the same training data used by MACE-OFF, using the pre-training, freezing and fine-tuning method described in Section 4.2, with the sGDML model as the prior.[14] To show that TTT is prior agnostic, we additionally train a model that uses the semi-empirical GFN2-xTB as the prior.[3] See A.4 for more details.

5.1.1 Force norm distribution shifts. Both MACE-OFF and GemNet-T deteriorate in performance when encountering systems with force norms different from those seen during training, as shown in Fig. 6. Interestingly, this performance drop occurs for both higher and lower force norms than those in the training set. Test-time training reduces errors for GemNet-T on out-of-distribution force norms, and also helps decrease errors for the new systems that are closer to the training distribution. The results in Fig. 6 specifically filter out atomic feature shifts and different connectivities to isolate the effect of force norm distribution shifts.

5.1.2 Connectivity distribution shifts. For both MACE-OFF and GemNet-T, force errors increase when the connectivity of a test graph differs from that of the training graphs, as measured by the spectral distance (see eqn (5)). Our test-time radius refinement (RR) technique (see Section 4.1) applied to MACE-OFF effectively mitigates connectivity errors at minimal computational cost. Test-time training also effectively mitigates connectivity distribution shifts, as shown in (Fig. 7 and Table 5). Note that Fig. 7 isolates connectivity distribution shifts by filtering out-of-distribution force norms and atomic features. See Section A.3.3 for RR results with the JMP model on the ANI-1x dataset.

5.1.3 Atomic feature distribution shifts. MACE-OFF and GemNet-T both perform poorly when encountering molecules with atomic features that differ from their training distributions. In particular, MACE-OFF and GemNet-T struggle with molecules that are both larger and smaller than those seen in training, and with systems that have a different proportion of carbon atoms than seen in training (see Fig. 11). Test-time training reduces errors across both of these atomic feature distribution shifts for GemNet-T. We filter out out-of-distribution connectivities and force norms to isolate the effect of atomic feature distribution shifts.

5.1.4 Aggregated results and takeaways. We present aggregated results on the SPICEv2 distribution shift benchmark, where a model is trained on SPICE and evaluated on 10k new molecules from SPICEv2. The large MACE-OFF universal model trains on 951k samples but still suffers from distribution shifts on the new structures from SPICEv2. We also see that (1) the RR method mitigates connectivity distribution shifts for MACE-OFF at minimal computational cost (see Table 1) and (2) using TTT with the GemNet-T model performs the best on the new molecules from SPICEv2, highlighting the effectiveness of

Table 1 Aggregated results on SPICEv2 distribution shift benchmark. We provide aggregated results on the SPICEv2 distribution shift benchmark with 95% confidence intervals. TTT and RR are both able to effectively mitigate errors across the 10k unseen molecules from SPICEv2. The relative improvements observed are in line with previous test-time training work[27,62]

Model	SPICEv2 test set force MAE (meV Å^{-1})
MACE-OFF	26.75 ± 0.65
+RR (ours)	26.0 ± 0.64
GemNet-T	22.9 ± 1.4
+TTT (ours)	**19.9 ± 1.0**

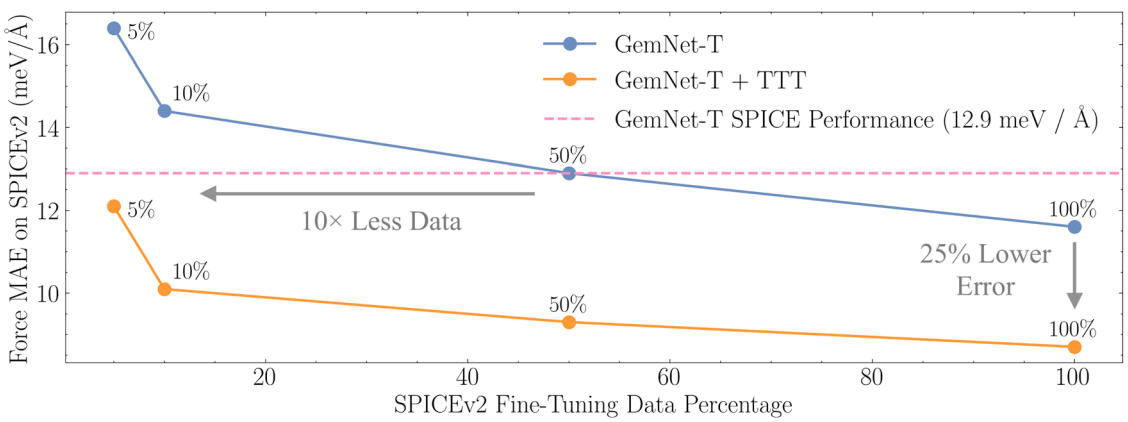


Fig. 5 Test-time training decreases the amount of fine-tuning data needed to match in-distribution performance. We fine-tune GemNet-T models, trained on SPICE, on new molecules from the SPICEv2 dataset. Applying TTT on the new data before fine-tuning decreases the amount of training data needed to match the in-distribution performance by 10×. Applying TTT before fine-tuning also decreases the final error by 25% when training on all the data.

training strategies for mitigating distribution shifts. Practically, these lower force errors also translate into better MD simulations and improved structure relaxations (see Section A.3.1).

Since the improvements from RR and TTT are right-skewed, meaning many molecules show small improvements while some see large gains, we highlight the 10% of molecules with the greatest improvement in Fig. 6b, 7b, and 11b. We also present results for individual molecules in Tables 4 and 5 to show that TTT and RR can help across a range of errors. Both TTT and RR improve results on molecules that already have low errors, and bring many molecules with high errors close to the in-distribution performance (see Fig. 12 which shows that more than 8 000/10 000 molecules have errors below 25 meV Å^{-1}).

The ability of TTT and RR to mitigate distribution shifts supports the hypothesis that MLIPs easily overfit to training distributions, even with large datasets. By improving the connectivity and learning more general representations of test molecules, RR and TTT diagnose the specific ways in which MLIPs overfit. These experiments suggest that improved training strategies, such as graph-free approaches,[73] could help learn more general models.

5.1.5 Test-time training and fine-tuning. While TTT can enable accurate MD simulations for new systems without access to any reference labels (see Section 5.2 and A.3.1), a practitioner may prefer to fine-tune a model on more out-of-distribution data to match the in-distribution performance. We examine how TTT can provide a better starting point for fine-tuning by learning more robust representations for new systems.

We take the GemNet-T models from the previous section and fine-tune them with varying amounts of structures from the SPICEv2 dataset.[21] We evaluate how much data is required to match the in-distribution force error on SPICE (12.9 meV Å^{-1}) when tested on the 10k new molecules from SPICEv2. The vanilla GemNet-T model matches the in-distribution performance when trained on half of the SPICEv2 data. In contrast, using our TTT procedure before fine-tuning allows the model to reach the same performance with only 5% of the data—a 10× reduction. Additionally, TTT reduces the final error by 25% even

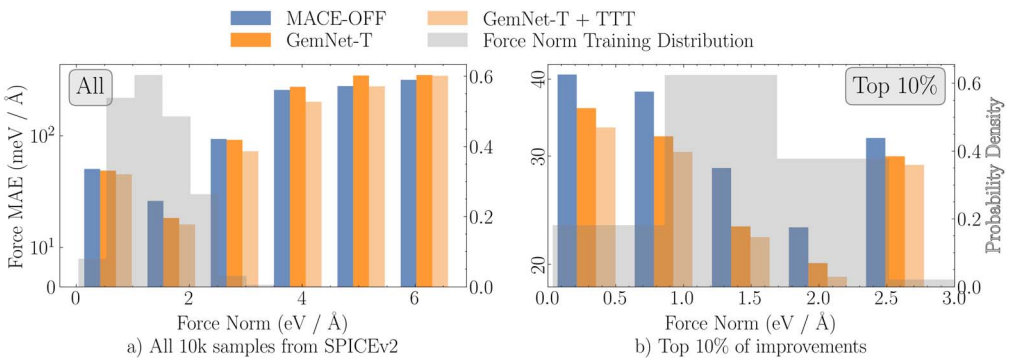


Fig. 6 Evaluating distribution shifts for force norms on SPICEv2. We evaluate MACE-OFF on new molecules from the SPICEv2 dataset with varying force norms. (a) Test structures with different force norms relative to the training distribution (shown in gray) incur larger force errors for MACE-OFF. We also train a GemNet-T model, and then apply test-time training (TTT), mitigating this shift. (b) We highlight the top 10% of molecules with the greatest improvement to demonstrate that TTT is effective even for structures that are near the training distribution.

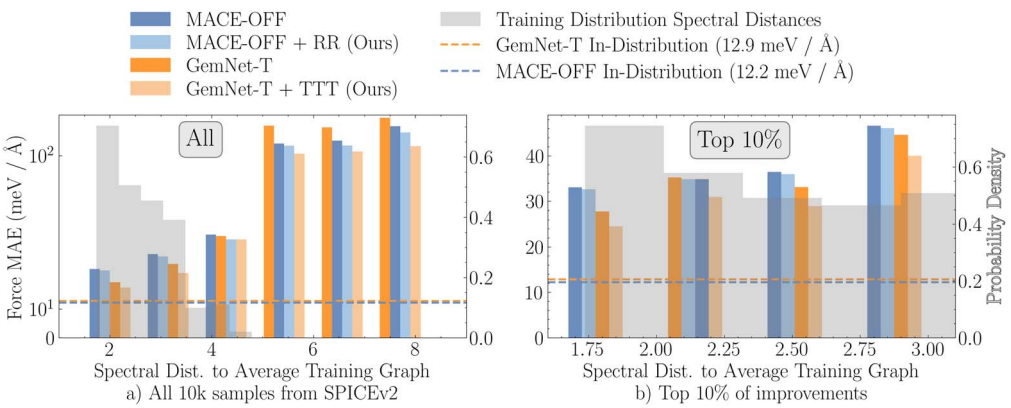


Fig. 7 Evaluating connectivity distribution shifts on SPICEv2. We evaluate MACE-OFF on new molecules from the SPICEv2 dataset with varying connectivity, defined by the spectral distance to the average training graph (see Section 4.1 for details). (a) Test structures with different connectivity relative to the training distribution (shown in gray) incur larger force errors for MACE-OFF. Test-time training (TTT) applied to a GemNet-T model and test-time radius refinement (RR) applied to MACE-OFF are both able to mitigate this performance drop at minimal computational cost. (b) We highlight the top 10% of molecules with the greatest improvement to demonstrate that TTT is effective even for connectivities close to the training distribution.

when fine-tuning on the entire SPICEv2 dataset. See Fig. 5 for details.

5.2 Evaluating generalization with extreme distribution shifts: simulating unseen molecules

We establish an extreme distribution shift benchmark to evaluate the generalization ability of MLIPs on the MD17 dataset.[13] This benchmark is specifically designed to highlight howMLIP training strategies tend to overfit to narrow problem settings, and to evaluate how new training strategies can improve robustness. We train a single GemNet-dT model[29] on 10k samples each of aspirin, benzene, and uracil. We then evaluate whether this model can simulate two new molecules, naphthalene and toluene, which were unseen during training. Next, we evaluate whether TTT can address the distribution shifts to the new molecules. Using the same procedure outlined in Section 4.2, we pre-train on the 3 molecules in the training set with the sGDML prior, then freeze the representation model and fine-tune on the quantum mechanical labels. We then perform TTT before simulating the new molecules (see Section 4.2). This is an extremely challenging generalization task for MLIPs due to the limited variety of training molecules. Nevertheless, we believe that a model capable of accurately capturing the underlying quantum mechanical laws should be able to generalize to new molecules.

We evaluate the stability of simulations over time by measuring deviations in bond length, following Fu *et al.*[24] We additionally calculate the distribution of interatomic distances $h(r)$, a low dimensional descriptor of 3D molecular structures, to measure the quality of the simulations.[24,52,69] See Section A.4 for more details.

5.2.1 Simulation results. As shown in Fig. 8, TTT enables stable simulations of unseen molecules that accurately reproduce the distribution of interatomic distances $h(r)$. Without TTT, the GemNet-dT model trained only on aspirin, benzene, and uracil is unable to stably simulate the new molecules and produces poor $h(r)$ curves. Even when we reduce the timestep by a factor of 5000 (from 0.5 fs down to 0.0001 fs), the simulations without TTT remains unstable. While the simplicity of the sGDML prior with its added inductive biases allows the prior to produce stable simulations, sGDML lacks the expressive power to capture the full details of the $h(r)$ curve. Test-time training on top of the prior

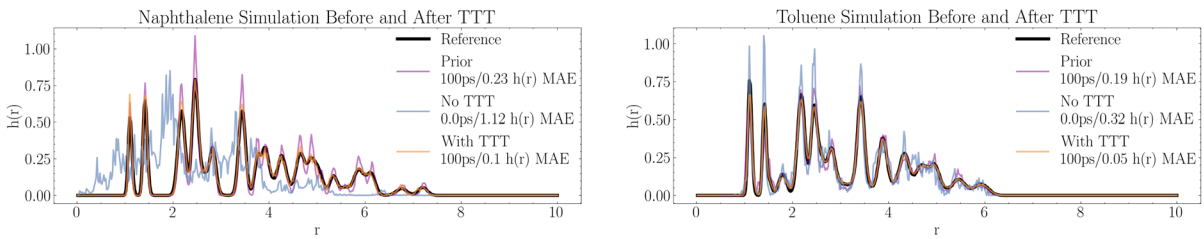


Fig. 8 Testing molecular dynamics simulations. TTT enables stable simulations that accurately reconstruct observables, such as the distribution of interatomic distances, for molecules not seen during training (orange). In contrast, predictions without TTT for these unseen molecules result in unstable simulations and inaccurate $h(r)$ curves (blue). Simulations without TTT remained unstable even with a timestep reduced by 5000× (from 0.5 fs to 0.0001 fs). We also show the predicted $h(r)$ from the sGDML prior. Since the prior is a simpler model with a number of physical inductive biases, it can produce stable simulations but lacks the expressive power to capture the full details of the $h(r)$ curve. TTT on top of the prior is able to better capture these nuances.

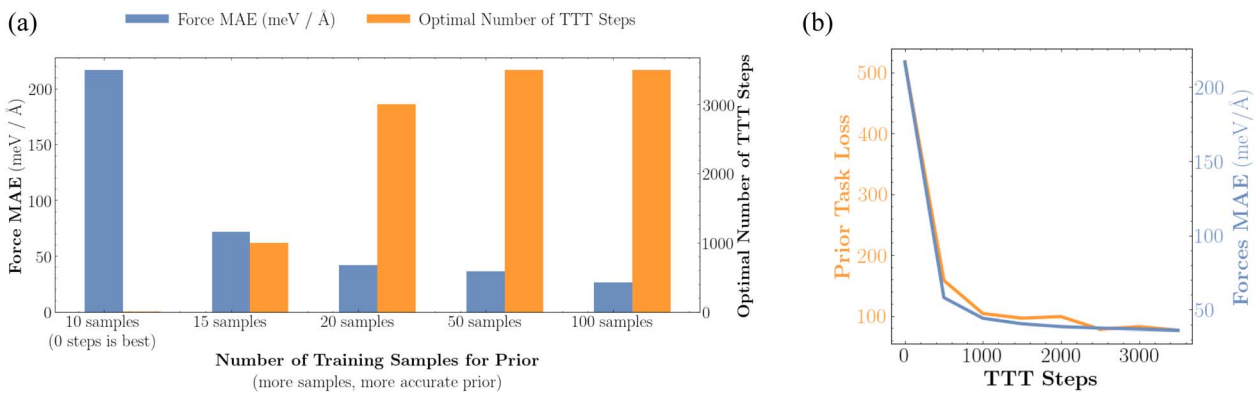


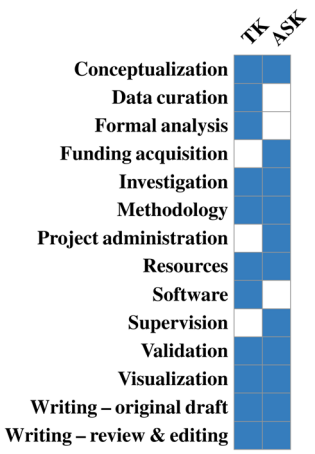
Fig. 9 Understanding the auxiliary task in TTT. We train a GemNet-dT model on three molecules from MD17 and perform TTT on naphthalene, a new molecule not seen during training. Our auxiliary objective for TTT is a cheap physical prior. We analyze how the accuracy of the prior affects the performance of TTT (a) and how the prior task loss relates to errors on the main task (b). (a) Impact of prior accuracy on test-time training (TTT) for naphthalene. As the prior becomes more accurate by training on more samples, we see larger improvements from TTT (blue bar). This accuracy allows us to take more gradient steps on the prior task (orange bar), without deteriorating performance on the main task. (b) Relationship between prior task loss and main task loss. Fitting to the prior task loss (orange) improves performance on the main task (blue) on naphthalene.

is able to better capture these nuances. A more accurate prior certainly can enable better performance after test-time training, but TTT is still possible with a weak prior (see Fig. 9a).We also find that TTT enables stable NVE simulations (see Section A.3.2). Furthermore, TTT provides a better starting point for fine-tuning, decreasing the amount of data needed to reach the in-distribution performance by more than 20× (see Section A.3.2). Given that GemNet-dT + TTT can produce reasonable simulations without access to quantum mechanical labels of the new molecules, test-time refinement methods could be a promising direction for addressing distribution shifts.

6 Conclusion

We have demonstrated that state-of-the-art universal MLIPs, even when trained on large datasets, suffer from predictable performance degradation due to distribution shifts. By identifying shifts in atomic features, force norms, and connectivity, we have developed methods to diagnose the failure modes of MLIPs. Our test-time refinement methods represent initial steps in mitigating these distribution shifts, showing promising results in modeling and simulating systems outside of the training distribution. These results provide insights into how MLIPs overfit, suggesting that while MLIPs are becoming expressive enough to model diverse chemical spaces, they are not being effectively trained to do so. This may indicate that training strategies, alongside data and architecture innovations, will be important in improving MLIPs. Finally, our experiments serve as benchmarks for evaluating the generalization ability of the next generation of MLIPs.

Author contributions



Conflicts of interest

There are no conflicts to declare.

Data availability

Data for this article are available through HuggingFace: https://huggingface.co/datasets/tkreiman/mlff_distribution_shifts (DOI: https://doi.org/10.57967/hf/6775). The code can be found on GitHub: https://github.com/ASK-Berkeley/MLFF-distribution-shifts (DOI: https://doi.org/10.5281/zenodo.17401634).

A Appendix

A.1 Details on test-time refinement training strategies

A.1.1 Test-time training (TTT). We elaborate on the details of our proposed test-time training (TTT) approach.

A.1.1.1 Model setup. Our model consists of the representation model, the main task head, and the prior task head, with parameters θ_R, θ_M, and θ_P respectively:

(1) The representation model, θ_R, is designed to extract features useful for both the main and prior task heads. These parameters can be trained on both the cheap data from the physical prior and the expensive reference calculations. After pre-training, the representation parameters can be further refined through fine-tuning and test-time training.

(2) The main task head, θ_M, predicts the energies and forces generated by DFT calculations. This head specifically uses the high-accuracy, expensive quantum mechanical labels produced by DFT for training.

(3) The prior head, θ_P, predicts the energies and forces from the cheap physical prior, such as classical force fields. This head is trained with the cheap labels produced by the physical prior.

We emphasize that the pre-training and test-time training procedures described in Section 4.2 are model architecture agnostic. For details on how we split up existing architectures into the representation model, main task head, and prior head, see Section A.4.

A.1.1.2 Necessity of proper pre-training for test-time training. The goal of TTT is to adapt to out-of-distribution test samples using a self-supervised objective at test-time.[27,42,62] In our case, we use the prior task loss $\mathcal{L}_P$ as the test-time training objective, making the model predict forces and energies labeled by the cheap physical prior. When an out-of-distribution (OOD) sample is encountered at test-time, we can adapt our representation parameters, θ_R, using the prior. This update improves the features extracted from the OOD samples, which in turn smooths the potential energy surface and improves the performance on the main task (see Fig. 9b). Importantly, naive fine-tuning of the full pre-trained model (both θ_R and θ_M) hinders the effectiveness of TTT. This is because fine-tuning θ_R on the main task may cause these parameters to "forget" the features learned from the prior during pre-training. If we adjust θ_R at test-time based solely on the prior targets, this could shift θ_R away from the representations that θ_M relies on to make predictions. Thus, for TTT to be successful, it is essential that the main task head depends on the features learned from the prior to make accurate predictions.

A.1.1.3 Notes on the prior. Although the performance of TTT does improve with a more accurate prior (see Fig. 9a), we note

that even in cases where the prior is poorly correlated with the main task (like with the EMT prior and OC20 in Section A.3.5), TTT still provides benefits. This is because the prior is only used to learn representations, and not to directly make predictions on the targets. This means that as long as training on the prior yields good representations, it can be used for TTT.

We also argue that such a prior is in fact widely available. For instance, one could always train an sGDML prior on the existing reference data. Alternatively, one could use a simple potential (like EMT or Lennard-Jones). A different (cheaper) level of quantum mechanical theory can also be used. Alternatively, as with prior TTT work in computer vision, a fully self-supervised objective (like atomic type masking and reconstruction) could also be used. We leave explorations of more priors to future work.

It should be noted that using sGDML as the prior requires a few labeled examples to train the sGDML model for the unseen molecule. We show that as few as 15 labeled examples are sufficient to tune the prior and achieve good TTT results (see Fig. 9a). TTT also yields better results than fine-tuning directly on these 15 samples, since the model severely overfits on the small number of samples. We also emphasize that across the board, TTT performs better than the prior (see Table 2). In addition, the sGDML prior only works on one system, whereas the MLIP can model multiple systems.

A.1.1.4 Limitations. Test-time training incurs extra computational cost, mainly due to the gradient steps taken at test time.

Table 2 Accuracy of prior for TTT. TTT always outperforms the prior

Molecule and number of training samples (or source)	Force MAE (meV Å^{-1})
Naphthalene	
10 samples	444.03
15 samples	123.98
20 samples	51.77
50 samples	42.28
100 samples	20.86
Toluene	
50 samples	44.82
Ac-Ala3-NHMe	
Ref. 15	34.25
Stachyose	
Ref. 15	29.05
Buckyball catcher	
100 samples	99.15
Average over 10k molecules from SPICEv2	
∼20 samples	62.25 (up to 724.5)
EMT	
Ref. 41	415
GFN2-xTB on SPICEv2	
Ref. 3	201.6

Table 3 RR maintains energy conservation. When using RR, we select the updated radius for the new molecule at the start of simulation and then keep it fixed. We update the envelope function to ensure smoothness of the predicted potential energy surface with the new radius. We run 10 ps NVE simulations to verify that RR does maintain a conservative force field

Model	Energy deviation (eV)
MACE-OFF	0.0036 ± 0.0004
MACE-OFF + RR	0.0049 ± 0.0022
GemNet-dT (non-conservative)	>1.0

This cost is negligible compared to the overall training time of a model, and negligible compared to the time it takes to run simulation with the model. Additionally, our instantiation of TTT requires access to a prior. However, a suitable prior is almost always available since one can always use a widely applicable analytical or semi-empirical potential.

A.1.2 Test-time radius refinement (RR). In this section we discuss further details about our RR approach (for theoretical justification, see Section A.2). Although one potential worry about using RR is that it might introduce potential discontinuities, we emphasize that we update the envelope function to ensure that the predicted potential energy surface remains smooth with the new radius. When running MD simulations, we choose the updated radius at the beginning and keep it fixed over the course of simulation. We verify that this maintains a conservative force field by running NVE simulations (see Table 3). Additionally, one might worry that the introduction of new edges will cause the model to overcount certain interactions. However, since edge features contain distance information, and since the model is trained on structures with varied edge distances, a well-trained model should be able to extract features from different edges. We note again that this is not an issue inherent to RR, since GNN-based MLIPs already deal with atoms entering a neighborhood during the course of simulation. Empirically, our experiments show that RR decreases force errors and improves simulation stability (see Section 5.1 and Table 5).

A.2 Theoretical motivation for test-time refinement

A.2.1 Test-time training. We provide theoretical justification for the intuition behind test-time training for machine learning interatomic potentials: if we have access to a cheap prior that approximates the reference labels, then taking gradient steps on the prior task will improve performance on the main task. Although making rigorous theoretical statements about deep neural networks in general is challenging, following previous test-time training works,[62] we assume a linear model to provide theoretical guarantees.

Theorem B.1 *(TTT with a Lennard-Jones Prior Improves Performance on Quantum Mechanical Predictions).*

Consider the linear model with representation parameters $R \in \mathbb{R}^{f\times d}$, *main task head parameters* $m \in \mathbb{R}^{d\times 1}$ *and prior task head parameters* $p \in \mathbb{R}^{d\times 1}$. *Main and prior task head predictions on input* $x \in \mathbb{R}^{f\times 1}$ *are given by* $\hat{E}^{\mathrm{P}} = x^T Rp, \hat{E}^{\mathrm{M}} = x^T Rm$. *Let* R_x' *be the updated*

representation weight matrix after one step of gradient descent on the prior loss with x as input, and learning rate η, and energy labels given by the Lennard-Jones potential:

$$R_x^{'} \leftarrow R - \eta \nabla_R \mathcal{L}_P\left(x^T R p, E^P\right) = R - \eta\left(E^P - x^T R p\right)\left(-x p^T\right).$$

If the reference energy calculations asymptotically go to ∞ as pairwise distances go to 0, and the features are chosen such that the activations (A = XR) have column rank d, then there exist inputs x such that:

$$\mathcal{L}_M\left(x^T R_x^{'} m, E^M\right) < \mathcal{L}_M\left(x^T R m, E^M\right).$$

In other words, taking gradient steps on the prior reduces the main task loss.

The proof builds on the main theoretical result presented by Sun *et al.*:[62]

Proof. Based on Sun *et al.*,[62] it suffices to show that there exist inputs x such that:

$$\text{sign}(E^P - x^T R p) = \text{sign}(E^M - x^T R m), \tag{9}$$

and

$$p^T m > 0. \tag{10}$$

In other words, the errors are correlated, and the task heads use similar features.

To see that there exist test points where the errors are correlated (eqn (9)), we use the fact that both the Lennard-Jones prior and the reference energies (by assumption) go asymptotically to ∞ as pairwise distances go to 0. Our linear model, however, can only make predictions within a bounded range over a bounded domain. Therefore, there clearly exists some x with pairwise distances small enough such that

$$x^T A p < E^P \text{ and } x^T A m < E^M,$$

implying that

$$(E^P - x^T A p),(E^M - x^T A m) > 0.$$

In other words, we can always find points where our model will underpredict both the prior and the main task energies.

To see that the task heads use similar features (eqn (10)), we consider a set $X \in \mathbb{R}^{n \times f}$ of n training examples. If we freeze the representation parameters as described in Section 4.2, then by least squares the learned p and m are:

$$p = (A^T A)^{-1} A^T y^P,\ m = (A^T A)^{-1} A^T y^M$$

where y^P, y^M are the vectors of prior and main task energies, respectively. Then:

$$p^T m = (y^P)^T A((A^T A)^{-1})^T (A^T A)^{-1} A^T y^M = (y^P)^T C y^M. \tag{11}$$

By the assumptions, we can express y^P, y^M in the orthogonal eigenbasis of C (with eigenvalues and eigenvectors λ_i, v_i):

$$y^P = \sum_j c_j v_j,\ y^M = \sum_k c_k v_k$$

Since we can always choose test-time training inputs where both the prior and the reference energy goes to ∞, then there clearly exist points where:

$$(y^P)^T y^M > 0, \tag{12}$$

implying that y^P, y^M share a common eigenvector with $c_j c_k > 0$.

Returning to eqn (11):

$$\begin{aligned}\left(y^P\right)^T C y^M &= \left(\sum_j c_j v_j{}^T\right) C \left(\sum_k c_k v_k\right) \\ &= \left(\sum_j c_j v_j{}^T\right)\left(\sum_k \lambda_k c_k v_k\right) > 0\end{aligned}$$

where the last inequality holds because of eqn (12) and the fact that C is positive definite.

To summarize, since the prior approximates the reference energies, we have shown we can find points where the errors are correlated and the model uses the same features. Using the theorem from Sun *et al.*,[62] this implies that gradient steps on the prior task improve performance on the main task, concluding the proof.

A.2.2 Test-time radius refinement. Our test-time radius refinement strategy is based on the theoretical finding presented by Bechler-Speicher *et al.*,[8] which states that GNNs tend to overfit to generally regular and well-connected training graphs. Although the theorems are presented for classification problems, they provide intuition and motivation for our RR approach. We restate some of the important theoretical results here (for the proofs and more details see Bechler-Speicher *et al.*[8] and Gunasekar *et al.*[34]).

Theorem B.2 (Extrapolation to new graphs[8]). *Let f* be a graph-less target function (it does not use a graph to calculate its output). In other words, f*(X, A) = f*(X), where X are node features and A is the adjacency matrix of a graph. There exist graph distributions P_1 and P_2, with node features drawn from the same fixed distribution, such that when learning a linear GNN with gradient descent on infinite data drawn from P_1 and labeled with f*, the test error on P_2 labeled with f* will be $\geq \frac{1}{4}$. In other words, the model fails to extrapolate to the new graph structures at test time.*

Mapping this to MLIPs, Theorem B.2 suggests that a GNN trained on specific types of molecular structures (*i.e.*, acyclic molecules) could fail to generalize to new connectivities at test time (*i.e.*, a benzene ring).

Theorem B.3 (Extrapolation within regular graph distributions[8]). *Let D_G be a distribution over r-regular graphs and D_X be a distribution over node features. A model trained on infinite samples from D_G, D_X and labeled by a graph-less target function f* will have zero test error on samples drawn from D_X, $D_{G'}$ (and labeled by f*), where $D_{G'}$ is a distribution over r′-regular graphs.*

In other words, generalizing across different types of regular graphs is easier for GNNs. Based on these theorems and our observation that many molecular datasets (MD17, MD22, SPICE) contain generally regular and well-connected graphs, we

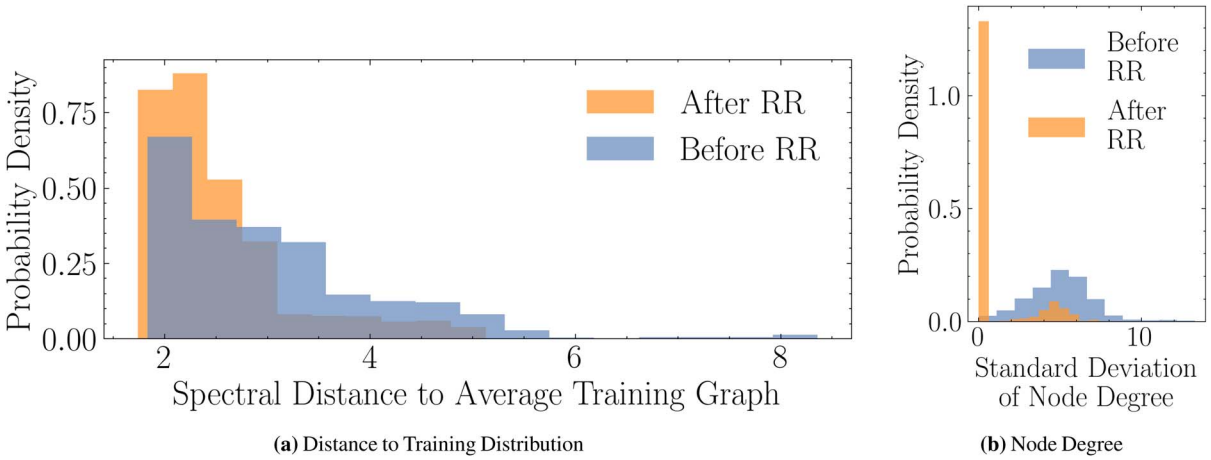


Fig. 10 Effect of radius refinement (RR) on molecular graph connectivities. We compare the connectivities of new molecular systems from the SPICEv2 dataset to the training distribution from SPICE, using the MACE-OFF training radius cutoff. Our results show that RR brings the connectivities of these molecular systems closer to the training distribution, as measured by the spectral distance (a) (note that for some molecular systems, the connectivity doesn't change unless the radius is made very small). Additionally, RR leads to more regular graph structures, with a reduced standard deviation of node degrees (b), indicating that the graphs are more regular.

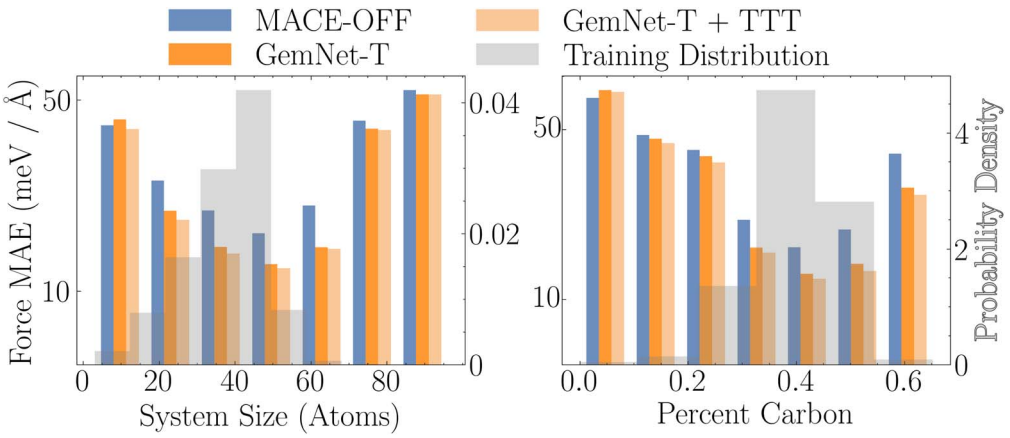


Fig. 11 Assessing the impact of atomic feature distribution shifts on model pperformance on SPICEv2 benchmark. We evaluate models trained on 951k samples from SPICE on new molecules from SPICEv2. The MACE-OFF model deteriorates in performance when encountering systems of different sizes or molecules with a different proportion of carbons atoms compared to its training set. We train a GemNet-T model on the 951k samples and run TTT—this is able to mitigate the atomic feature distribution shifts.

are motivated to find ways to make testing graphs look more like the training distribution (generally regular and well-connected) to help the models generalize. The observation that graphs for MLIPs are often generated by a radius cutoff led us to develop the RR method presented in Section 4.1. See Fig. 10, which empirically shows that RR makes graphs more regular and brings them closer to the distribution of training connectivities, aligning with our theoretical intuition. While we think it is an interesting direction for future research to continue exploring the theoretical properties of graph structure distribution shifts.

A.3 Additional test-time refinement results

We provide additional test-time refinement experiments using more models, datasets, and priors. Although these constitute challenging generalization tasks, test-time refinement shows promising first steps at mitigating distribution shifts and generalizing to new types of systems.

A.3.1 Further results on SPICEv2 distribution shift benchmark. Since the TTT and RR results for the SPICEv2 distribution shift benchmark (see Section 5.1) are right skewed, there are many molecules that only improve slightly and a few that improve dramatically. In Tables 4 and 5, we highlight results from 6 randomly selected molecules from the top 1000 most improved with TTT and RR. Specifically, two molecules were randomly chosen from each of the following force error bins: 0––40, 40––100, and >100 meV $Å^{-1}$. These results show that TTT and RR help across a range of errors: bringing high errors down to below 40 meV $Å^{-1}$, and improving results on already low errors.

We also explicitly quantify in Fig. 12 that many molecular systems start with large errors and these errors are decreased to well within 40 meV $Å^{-1}$ with TTT and RR. Additionally,

Table 4 Benefit of test-time training (TTT). We evaluate a GemNet-T model trained on 951k samples from SPICE on 10k new molecules from SPICEv2. We highlight specific examples from SPICEv2 where TTT provides large improvements. TTT can decrease errors by an order of magnitude, and can bring errors close to in-distribution performance. Even when errors are already low, TTT can further reduce errors. TTT also improves NVT simulation stability (mean ± standard deviation reported over 3 seeds)

	C_4NH_{12}	$N_3C_5H_3$	IC_2H	$ClOC_{14}NH_{15}$	$C_{10}N_2C_3H_{14}$	O_3P
GemNet-T	28	18	93	55	210	748
Force MAE (meV Å^{-1})/stability (ps)	100 ± 0	100 ± 0	14.7 ± 1.2	100 ± 0	100 ± 0	18.5 ± 0.7
GemNet-T + TTT	16	13	42	31	70	91
Force MAE (meV Å^{-1})/stability (ps)	100 ± 0	100 ± 0	38.2 ± 6.0	100 ± 0	100 ± 0	100 ± 0

Table 5 Benefit of radius refinement (RR). We evaluate MACE-OFF, trained on 951k samples from SPICE, on 10k new molecules from SPICEv2. We highlight specific molecules from SPICEv2 to show that RR improves errors across a range of values. RR also improves NVT simulation stability (mean ± standard deviation reported over 3 seeds)

	IC_2H	$O_5N_3C_{16}H_{35}$	$N_4C_7H_{11}$	$O_4C_2PH_6$	$C_6N_2H_{12}$	SC_6H_4
MACE-OFF	23	12	58	79	875	109
Force MAE (meV Å^{-1})/stability (ps)	100 ± 0	38.7 ± 12.6	100 ± 0	100 ± 0	62.8 ± 26.3	100 ± 0
MACE-OFF + RR	16	9	39	49	374	69
Force MAE (meV Å^{-1})/stability (ps)	100 ± 0	78.9 ± 16.3	100 ± 0	100 ± 0	100 ± 0	100 ± 0

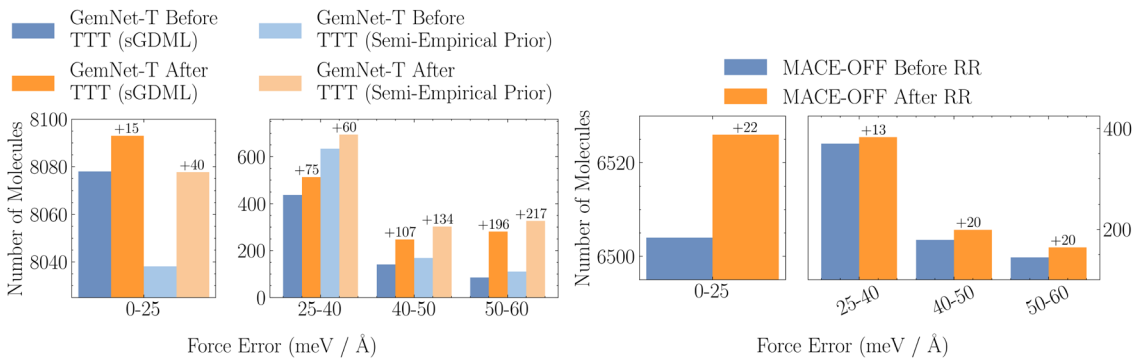


Fig. 12 Test-time training and radius refinement strategies for improved molecular force prediction. We train a GemNet-T model (left) on 951k samples from the SPICE dataset and evaluate it on new molecules from the SPICEv2 dataset. We also evaluate the MACE-OFF model (right), which was also trained on the same 951k samples from SPICE. We plot the number of molecules that fall into specific force error bins to show that TTT (left) and RR (right) help improve errors for hundreds of molecular systems. As with previous test-time training works, improvements are more challenging to achieve for systems with lower initial errors (*i.e.*, those closer to in-distribution performance), but TTT and RR still help bridge the gap to in-distribution performance.

hundreds of molecules across a range of errors have errors that are brought down significantly closer to the in-distribution performance. These results suggest that MLIPs have the expressivity to model more diverse chemical spaces, and can be better trained to do so.

A.3.1.1 Evaluating improved downstream utility. We run reference DFT simulations to ensure that the improved stability reported in Tables 4 and 5 translate into improved simulation quality in terms of the predicted distribution of interatomic distances $h(r)$ (see Sections 5.2 and A.4 for details). Due to the computational cost of running reference MD simulations, we are only able to do this on a subset of the molecules. RR on top of MACE-OFF lowers the $h(r)$ MAE from 0.17 and 0.09 to 0.15 and 0.07 for SC_6H_4 and IC_2H, respectively. TTT on top of GemNet-T lowers the $h(r)$ MAE from 0.20, 0.44, and 0.39 to 0.18, 0.19, and 0.17 for $N_3C_5H_3$, IC_2H, and O_3P, respectively.

We run also BFGS structure relaxations with the MLIPs and calculate how many extra steps are needed with DFT to find a relaxed structure. GemNet-T requires 10.4 additional steps on average (on $N_3C_5H_3$, IC_2H, and O_3P, 3 seeds each), whereas GemNet-T + TTT requires 7.7. MACE-OFF requires 4.0, and MACE-OFF + RR requires 3.8 (on SC_6H_4 and IC_2H, 3 seeds each).

Empirically, on individual molecular systems from Table 4, we find that the improved simulations are sometimes a result of improved modeling of non-bonded interactions: the model before TTT is sometimes unable to keep separate molecular fragments together. Paired with the qualitative smoothing of the potential energy surface, these results suggest that TTT is in fact improving the underlying representations for chemical systems. While it is in general hard to interpret the representations of deep neural networks, we think it is an interesting direction for future work to understand what inductive biases MLIPs learn and how TTT improves them.

Table 6 Test-time training (TTT) with a semi-empirical prior on SPICEv2 benchmark. We evaluate a GemNet-T model trained on 951k samples from SPICE on a held-out set of 10k new molecules from SPICEv2. To evaluate the effectiveness of TTT, we use the semi-empirical GFN2-xTB[3] as a prior and apply TTT to our SPICEv2 distribution shift benchmark. The results show that TTT with a semi-empirical prior improves performance across a range of error levels, bringing many molecules close to the performance achieved on in-distribution data. We report 95% confidence intervals for the overall error on the entire test set and highlight individual molecule examples to illustrate the benefits of TTT

		Overall	O_2ClSNC_8–H_{16}	$O_2N_2C_{16}$-SH_{14}	O_3C_{19}–SiH_{26}	$O_2N_2C_{16}$–SiH_{28}	Cl_2C_7–SiH_{14}	Cl_3C_9–SiH_{11}
Force MAE (meV Å^{-1})	GemNet-dT	78.3 ± 7.8	38	33	74	75	109	107
	GemNet-dT + TTT	56.6 ± 5.6	28	26	35	39	46	44

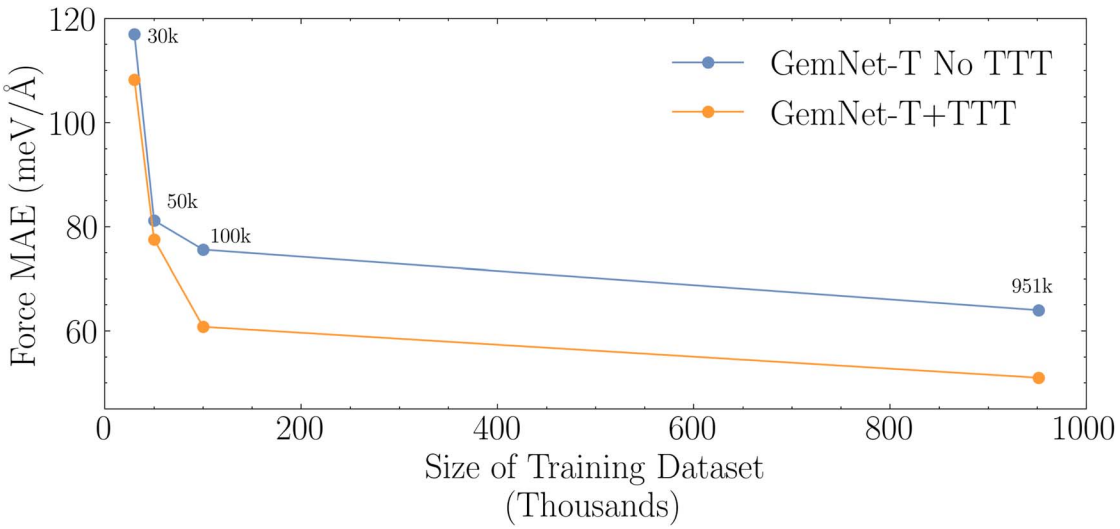


Fig. 13 Performance on the SPICEv2 distribution shift benchmark *versus* dataset size. We evaluate GemNet-T models trained on increasing amounts of data from SPICE on 10k new molecules from SPICEv2. The results show that while increasing the training dataset size improves performance on the SPICEv2 benchmark, the gains in accuracy diminish rapidly. Test-time training (TTT) consistently improves performance across all dataset sizes.

A.3.1.2 TTT is agnostic to the chosen prior. We explore using the semi-empirical GFN2-xTB[3] as the prior to provide further evidence that TTT is agnostic of the prior chosen. We train a GemNet-dT model with the pre-train, freeze, fine-tune approach described in Section 4.2 using GFN2-xTB as the prior. The results in Table 6 show that TTT with GFN2-xTB also enables better performance across a range of errors.

A.3.1.3 Scaling experiment on SPICEv2: investigating the impact of dataset size on out-of-distribution performance. We conduct a scaling experiment to understand out-of-distribution performance with and without TTT as a function of dataset size. We train four GemNet-T models on different subsets of the SPICE dataset: 30k, 50k, 100k, and the full 951k samples. Our results, presented in Fig. 13, show that increasing the dataset size improves generalization performance on SPICEv2, but with diminishing returns. This suggests that simply adding more in-distribution data may not be sufficient to achieve optimal generalization performance, consistent with our findings in Fig. 2 and Section 3. Notably, TTT consistently improves performance across all dataset sizes, and the benefits of TTT do not decrease even when using the full 951k dataset.

A.3.2 Additional results on MD17. We additionally run NVE simulations[24,25] with the Velocity Verlet integrator[39] before and after TTT. As with the NVT simulations, we use a 0.5 fs time step and simulate for 100 ps. Although simulations on naphthalene are slightly more unstable, TTT still increases the stability of simulations (see Table 7).

We also demonstrate that TTT can be used in conjunction with fine-tuning. We fine-tune the GemNet-dT model used in Section 5.2 on the out-of-distribution toluene molecule. We measure how much data is needed to reach the in-distribution performance of less than 15 meV Å^{-1}. This fine-tuning is done both before and after TTT is conducted. Fig. 14 shows that TTT provides a much better starting point for fine-tuning, reducing the number of reference labels needed to reach the in-distribution performance by more than 20×.

Table 7 Stability of NVE simulations with test-time training (TTT). We train a GemNet-dT model on three molecules from MD17 and evaluate its ability to simulate new molecules not seen during training. TTT enables stable NVE simulations for molecules unseen during training. We report mean ± standard deviation across 3 seeds

Molecule	GemNet-T	GemNet-T + TTT
Toluene	<1 ps	100 ± 0 ps
Naphthalene	<1 ps	43 ± 5.2 ps

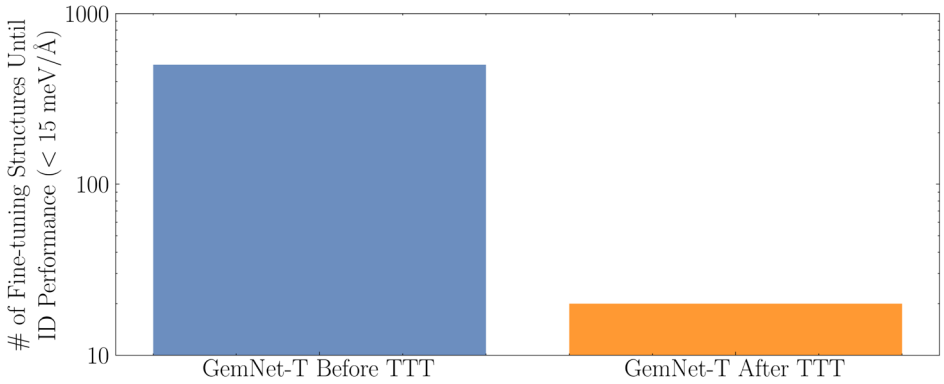


Fig. 14 Test-time training (TTT) improves fine-tuning efficiency on MD17 dataset. We demonstrate the effectiveness of TTT in reducing the amount of data required for fine-tuning a GemNet-dT model to achieve in-distribution performance. Initially, we train the model on a small set of three molecules from the MD17 dataset. We then fine-tune the model on a new, unseen molecule (toluene) with and without TTT. Our results show that applying TTT before fine-tuning enables the model to reach in-distribution performance (<15 meV Å^{-1}) with 10 times less data compared to fine-tuning without TTT.

As an example of a real-world force norm distribution shift, we run an additional MD simulation at 700 K on aspirin using the model from 5.2. Although the model saw aspirin during training, the MD17 data was generated with NVT simulations at 500 K. Consequently, running NVT simulations at a higher temperature will sample higher force norms more often than seen in the training data. We evaluate the quality of the simulations by again calculating the predicted distribution of interatomic distances $h(r)$. We use importance sampling[52] to get the reference $h(r)$ at 700 K. TTT increases stability from 6.5 to 53.2 ps and decreases $h(r)$ MAE from 0.17 to 0.10 when running simulations at 700 K (compared to 72 ps and 0.04 at 500 K).

A.3.3 Test-time radius refinement with JMP on ANI-1x. We evaluate whether our proposed test-time radius refinement (RR) method (see 4.1) can help JMP[58] address connectivity distribution shifts in the ANI-1x dataset.[60] Following the approach outlined in Section 5.1, we search over 7 different radius cutoffs from 6.5 to 9.5 Å to find the one that best matches the training Laplacian eigenvalue distribution.

As shown in Tables 8 and 9, RR is able to improve force errors for JMP, including improving errors that are already low. We again highlight the top 10% of molecules with the greatest improvement, since the improvements from RR are right-skewed. RR often improves errors by 10–20% for individual molecules. This experiment provides further evidence that RR can address connectivity distribution shifts for existing pre-trained models at minimal computational cost, suggesting that existing models overfit to the graph structures seen during training.

A.3.4 Evaluating distribution shifts in the MD22 dataset: low to high force norms. We establish a benchmark for force norm distribution shifts, using the MD22 dataset.[15] The MD22 data set contains large organic molecules with samples generated by running constant-temperature (NVT) simulations,

Table 8 Test-time radius refinement with JMP on ANI-1x. We implement our test-time radius refinement method (see Section 4.1) on JMP and evaluate improvements on the ANI-1x test set defined in Shoghi *et al.*[58] Test-time radius refinement helps improve performance by mitigating connectivity distribution shifts. We highlight the top 10% of molecules with the greatest improvement in parentheses to show that test-time radius refinement helps across a range of errors

	Force error range (meV Å^{-1})		
	0–43	43–100	>100
JMP on ANI-1x Test set (top 10%)	17.4 ± 0.02	52.4 ± 0.18	151.7 ± 8.4
Force MAE (meV Å^{-1})	(15.1 ± 0.07)	(52.3 ± 0.54)	(167.7 ± 39.3)
JMP + RR (ours) on ANI-1x Test set (top 10%)	17.3 ± 0.02	52.3 ± 0.18	151.5 ± 8.3
Force MAE (meV Å^{-1})	(14.6 ± 0.07)	(51.9 ± 0.54)	(163.6 ± 37.8)

Table 9 Individual examples from ANI-1x with radius refinement (RR) on JMP. We perform RR when evaluating JMP on molecules from the ANI-1x test set. We highlight individual molecular examples to show that RR helps across a range of errors

Example molecules force MAE before → after RR (meV Å^{-1})					
$C_3H_{10}N_2O_2$	C_5H_3NO	$C_5H_6N_2O$	$C_5H_5NO_2$	$C_5H_3N_3$	$C_3H_6O_2$
6.9 → 5.4	8.2 → 6.2	53.0 → 44.2	85.2 → 78.3	101.1 → 99.7	158.9 → 149.7

Table 10 Evaluating low to high force norms on MD22. We train a GemNet-dT model on low force norm structures from MD22 (<1.7 eV Å^{-1} force norm averaged over atoms) and evaluate the model on high force norm structures (>1.7 eV Å^{-1}). GemNet-dT generalizes poorly to the high force norm structures, but TTT significantly closes the gap

		Force MAE (meV Å^{-1})		
Force norm average	Model	Ac-Ala3-NHMe	Stachyose	Buckyball catcher
<1.7 eV Å^{-1}	GemNet-dT	11.6	11.7	8.7
>1.7 eV Å^{-1}	GemNet-dT	36.8	24.2	16.4
	↓	↓	↓	↓
	GemNet-dT + TTT	26.5	19.0	12.7

meaning that the majority of the structures are in lower energy states, and thus have low force norms. We filter out structures that have an average per-atom force norm smaller than a 1.7 eV Å^{-1} cutoff, which filters out about half of the data (Fig. 17). We then evaluate whether GemNet-dT can generalize to high-force norm structures.

We train three different GemNet-dT models on 3 MD22 molecules—Ac-Ala3-NHMe, stachyose, and buckyball catcher—using the filtered low force norm dataset. We evaluate the GemNet-dT model on structures with force norms larger than the training cutoff. We also perform TTT using sGDML as the prior, as described in Section 4.2, to mitigate the distribution shift on the high-force norm test samples. For more details, see Section A.4.

A.3.4.1 Force norm generalization results. As shown in Table 10, GemNet-dT performs poorly on high force norm structures when compared to the low force norm structures it sees during training. TTT can mitigate the force norm distribution shift and close the gap between the in-distribution and out-of-distribution performance. This result further supports the hypothesis that MLIPs struggle to learn generalizable representations even when facing a distribution shift in a narrow single molecule dataset.

A.3.5 Test-time training on OC20. The Open Catalyst 2020 (OC20) dataset consists of relaxation trajectories between adsorbates and surfaces.[12] The primary training objective consists of mapping structures to their corresponding binding energy and forces (S2EF), as determined by DFT calculations. Both the S2EF task and OC20 dataset are challenging, due to the diversity in atom types and system sizes. The OC20 dataset includes an out-of-distribution test split consisting of systems that were not encountered during training. Even models trained on the full 100M+ OC20 dataset perform significantly worse on the out-of-distribution split.[12] Consistent with previous test-time training work,[27,42,62] we use this split to assess our TTT approach.

A.3.5.1 Problem setup. For our prior, we use the Effective Medium Theory (EMT) potential, introduced by Jacobsen *et al.*[41] Using this, we can compute energies and forces for thousands of structures in under a second using only CPUs.[39] The EMT potential currently only supports seven metals (Al, Cu, Ag, Au, Ni, Pd and Pt), as well as very weakly tuned parameters for H, C, N, and O. Consequently, we filter the 20 million split in the OC20 training dataset to only the systems with valid elements for EMT, leaving 600 thousand training examples. Similarly, the validation split is filtered and reduced to 21 thousand examples. While this work primarily focuses on evaluating our TTT approach, exploring the potential of a more general prior, or developing such a prior, represents a promising direction for future work.

A.3.5.2 Training procedure. We use a joint training loss function, $\mathcal{L} = \mathcal{L}_P + \mathcal{L}_M$, to train a GemNet-OC model,[30] which is specifically optimized for the OC20 dataset. At test-time, we use the EMT potential to label all structures with forces and total energies. For each relaxation trajectory in the validation dataset, we update our representation parameters with the prior objective, $\mathcal{L}_P$ (see eqn (7)), and then make predictions with the updated parameters (see eqn (8)). The TTT updates are performed individually for each system in the validation set. See Table 12 for hyperparameters.

A.3.5.3 Results. We compare the performance of our joint-training plus TTT method against a baseline GemNet-OC model trained only on DFT targets and evaluated without TTT on the validation set. Despite the weak correlation between EMT labels and the more accurate DFT labels (see Fig. 15), using EMT labels for joint-training helps regularize the model and improves performance on the out-of-distribution split. After joint-training, implementing test-time training steps further

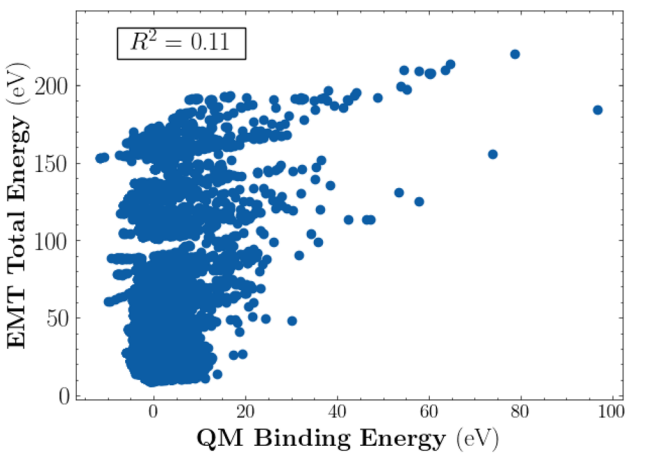


Fig. 15 EMT correlation with reference energy DFT calculations on OC20. We compare the DFT energy to the predicted energy from the EMT prior on samples from OC20. The correlation is very weak.

Table 11 OC20 test-time training. We evaluate a GemNet-OC model on the OC20 out-of-distribution validation split to assess the impact of joint-training and TTT. The model is trained on 600 thousand examples from the OC20 20M split that have elements supported by the EMT prior

Model	Force MAE (meV Å^{-1})	Energy MAE (meV)
GemNet-OC	77.8	1787.4
GemNet-OC joint training (ours)	63.67	1320
GemNet-OC joint training + TTT (ours)	**61.42**	**1143**

Table 12 TTT hyperparameters for OC20 OOD split

Hyperparameter	Value
Steps	11
Learning rate	1×10^{-4}
Optimizer	Adam
Weight decay	0.001

improves the model's performance (see Table 11). This demonstrates that even though EMT has limited predictive accuracy as a prior, it can still be used to learn more effective representations that generalize to out-of-distribution examples. This experiment provides further evidence that improved training strategies can help existing models address distribution shifts.

A.3.6 Additional potential energy surfaces before and after test-time training. We provide additional potential energy surface plots in Fig. 16. TTT consistently smooths the predicted potential energy surface. We plot the energy along the two principal components of the energy Hessian.

A.4 Experiment details

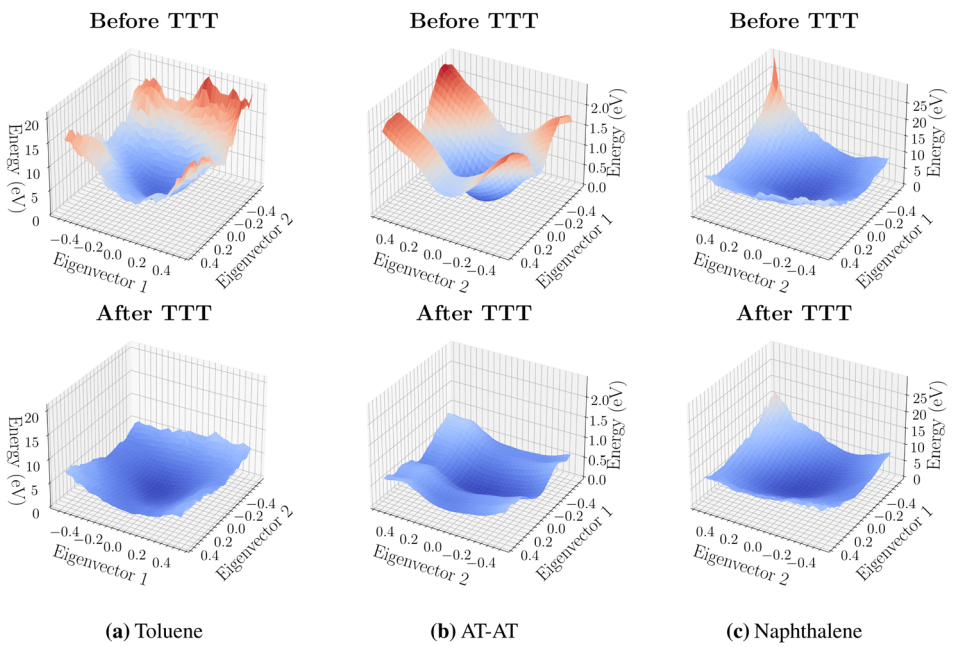
We describe in detail the benchmarks established in this paper along with experiment hyperparameters. Code for benchmarks and training methods will be made available.

In line with previous test-time training works,[27,42,62] we update as few parameters as possible during TTT. For MD17, MD22, and SPICE experiments, we train everything before the second interaction layer in GemNet-T/dT. For OC20 (see Section A.3.5), we train everything before the second output block in GemNet-OC.

Hyperparameters were largely adapted from Fu *et al.*,[24] although we increased the batch size to 32 to speed up training for GemNet-dT. Other deviations from Fu *et al.*[24] are mentioned below.

A.4.1 SPICEv2 distribution shift benchmark

A.4.1.1 Dataset details. We evaluate models trained on MACE-OFF's training split,[46] consisting of 951k structures primarily from the SPICE dataset.[20] The test set contains 10 000 new molecules from SPICEv2 (ref. 21) not seen in the MACE-OFF training split. The 10 000 molecules were chosen to be the molecules that had the most structures in order to provide



(a) Toluene **(b)** AT-AT **(c)** Naphthalene

Fig. 16 Predicted potential energy surfaces for molecules in MD17 and MD22. We consider a GemNet-dT model trained on three molecules from MD17 ((a) toluene, (b) AT–AT, (c) naphthalene). We plot the predicted potential energy surface, before and after test-time-training, from the model along the first two principal components of the Hessian for new molecules not seen during training. TTT regularizes the model and smooths the predicted potential energy surface.

Table 13 TTT parameters for SPICEv2 distribution shift benchmark

Parameter	Value
Learning rate	1×10^{-4}
Momentum	0.9
Optimizer	SGD
Weight decay	0.001
Steps	250

a large test set of 475 761 structures. GemNet-T was trained on the same data as MACE-OFF.

A.4.1.2 Simulation details. We run simulations for 100 ps using a temperature of 500 K and a Langevin thermostat (with friction 0.01), otherwise following the parameters used in Fu *et al.*[24] Since the SPICEv2 dataset was not generated purely from MD simulations, we do not have reference $h(r)$ curves for this dataset and instead focus on stability.

A.4.1.3 Hyperparameters. Hyperparameters were adapted from Fu *et al.*,[24] with the following modifications shown to scale the model to 4M parameters to be more in line with MACE-OFF's 4.7M parameters:

(1) Atom embedding size: 128 → 256
(2) RBF embedding size: 16 → 32
(3) Epochs: 250

For test-time training parameters, see Table 13. Note that we performed early stopping if the prior loss got stuck, or if it reached the in-distribution loss (since this implies overfitting and deteriorates performance on the main task).

A.4.2 Assessing low to high force norms on MD22

A.4.2.1 Dataset details. We train on approximately 6k samples from each molecule, corresponding to the 10% split for Ac-Ala3-NHME, 25% for stachyose, and 100% for buckyball catcher.

A.4.2.2 Hyperparameters. See Table 14 for details on the hyperparameters used.

A.4.3 Simulating unseen molecules on MD17. We provide further experimental details for the simulating unseen molecules benchmark on MD17 (see Section 5.2).

Table 14 TTT hyperparameters MD22 experiments. We note that especially in cases where the prior is reasonably accurate, TTT is generally robust to a wide range of hyperparameter choices

Hyperparameter	Value
Steps	50
Learning rate	1×10^{-5}
Optimizer	SGD
Momentum	0.9
Weight decay	0.001

A.4.3.1 Dataset details. We use the 10k dataset split for the 3 training molecules (aspirin, benzene, and uracil). For test-time training, the 1k test-set is used for naphthalene and toluene. We note that TTT can also be done with structures generated from simulations with the prior, and we think further experimentation with this is an interesting direction for future work.

A.4.3.2 Simulation details. We run simulations for 100 ps using a 0.5 fs timestep, a temperature of 500 K, and a Langevin thermostat (with friction 0.01 fs^{-1}), otherwise following the parameters used in Fu *et al.*[24] We measure the distribution of interatomic distances $h(r)$ to evaluate the quality of the simulations. The distribution of interatomic distances is defined as:

$$h(r) = \frac{1}{n(n-1)} \sum_{i}^{n} \sum_{j \neq i}^{n} \delta\left(r - \|\mathbf{x}_i - \mathbf{x}_j\|\right), \tag{13}$$

where r is a reference distance, x_i denotes the position of atom i, n is the total number of atoms, and δ is the Dirac Delta function. The MAE between a predicted $\hat{h}(r)$ and a reference $h(r)$ is given by:

$$\text{MAE}\left(\hat{h}(r), h(r)\right) = \int_0^\infty \left| \langle h(r) \rangle - \left\langle \hat{h}(r) \right\rangle \right| \text{d}r, \tag{14}$$

where $\langle \cdot \rangle$ indicates time averaging over the course of the simulation.

In both cases, TTT brings down force errors from ~200 meV Å^{-1} down to less than 25 meV Å^{-1}, beating the prior (that uses 50 samples) and enabling stable simulation. We found that

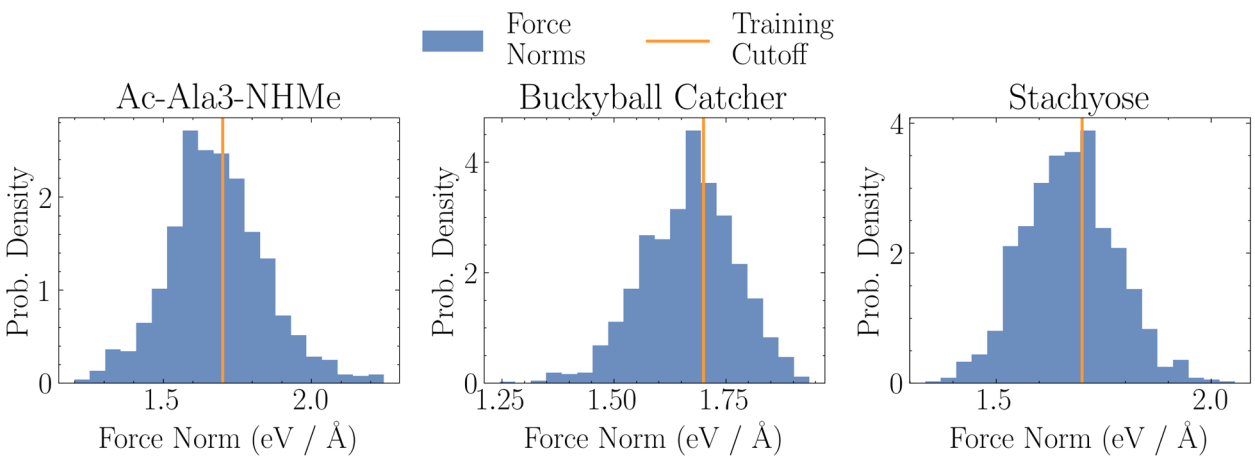


Fig. 17 Force norms for MD22 force norm distribution shift experiment. We plot the force norms for molecules from the MD22 dataset. The line in orange indicates the force norm cutoff used to train the models in Section A.3.4. Note that since the dataset was generated with NVT simulations, force norms are generally low when compared to SPICE.

Table 15 TTT parameters for MD17 transferability benchmark

Parameter	Value
Learning rate	1×10^{-3}
Momentum	0.9
Optimizer	SGD
Weight decay	0.001
Steps	3000

a prior that uses only 15 samples still leads to improvements with TTT (see Fig. 9a).

A.4.3.3 Hyperparameters. See Table 15 for hyperparameters used in the MD17 simulation experiments.

A.5 Details on distribution shifts

We emphasize that atomic feature, force norm, and connectivity distribution shifts define "orthogonal" directions along which a shift can happen in the sense that they can each happen independently. In other words, a structure might have the same connectivity and similar force norms, but have a different composition of elements. Similarly, for the SPICEv2 dataset, the distribution of connectivities is the same independent of force norm of the structure (see Fig. 20). This implies that one can observe a force norm shift while still seeing similar elements and connectivity.

Additionally, we provide more details on how we diagnose distribution shifts for new molecules at test time.

(1) Identifying distribution shifts in the atomic features $\mathbf{z}$ is straightforward: one can simply compare the chemical formula of a new structure to the elements seen during training.

(2) To diagnose force norm distribution shifts, we observe that although priors often have large absolute errors compared to reference calculations, force norms are actually highly correlated between priors and reference values (see Fig. 18 for an example from MD17). To determine whether a structure might be out-of-distribution with respect to force norms, the prior can be quickly evaluated at test time, and the predicted force norm can be compared to the training distribution.

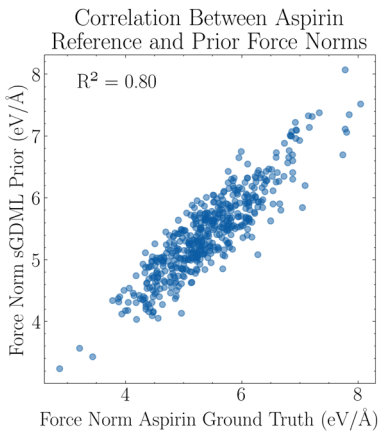


Fig. 18 Prior and reference force norms are highly correlated. We plot force norms calculated by the sGDML prior and the reference DFT for samples of aspirin from the MD17 dataset. The force norm predicted by the prior is highly correlated with the reference force norm, despite the absolute error between them being large.

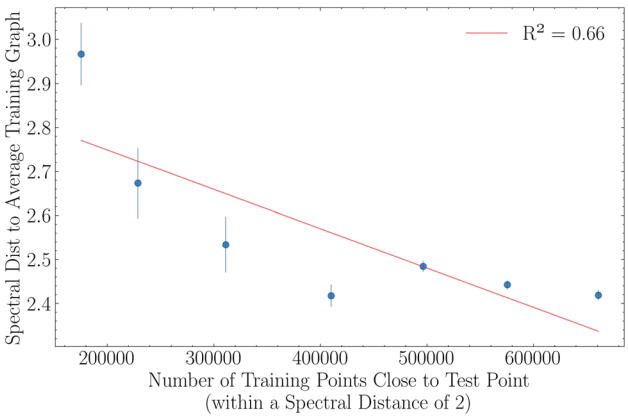


Fig. 19 Spectral distance to average training graph correlates with number of training samples close to test example. We compare the connectivity of new samples from the SPICEv2 dataset to those seen during training on the SPICE dataset. Although representing the training connectivities with an average Laplacian spectrum is lossy, comparing a test graph to this average spectrum correlates strongly with counting the number of training graphs close to the test graph. 95% confidence intervals are shown with error bars.

(3) Connectivity distribution shifts can be quickly identified by comparing graph Laplacian eigenvalue distributions with the spectral distance (see 4.1). Although comparing to the average Laplacian spectra is a lossy representation of the training distribution, comparing individually to all the training graphs is prohibitively expensive in practice. We also observe that counting the number of training graphs close to a test point correlates strongly with the spectral distance between the test graph and the average spectrum (see Fig. 19).

We emphasize that our proposed methods for diagnosing distribution shifts are computationally efficient, and they do not require access to reference labels.

A.6 Computational usage

All of our experiments were run on a single A6000 GPU.

- MD17/22: training for 100 epochs on a single molecule takes 2 GPU hours. Option 2 from Fig. 4a (pre-training, freezing, then fine-tuning) took 2 hours for pre-training and then 2 hours for fine-tuning (although we observed strong finetuning results with even less pre-training). TTT took less than 15 minutes for each molecule.
- SPICE results: pre-training on the prior took less than 5 hours on an A6000 across model sizes. Fine-tuning took 2 days. TTT took less than 5 minutes per molecule. In comparison, MACE-OFF small, medium, and large trained for 6, 10, and 14 A100 GPU-days respectively. Radius refinement takes less than 1 minute per molecule (to calculate eigenvalues to find the optimal radius).
- OC20: joint-training (option 1) took 48 hours. Evaluation with TTT took 6 hours (compared to 2 hours without TTT).

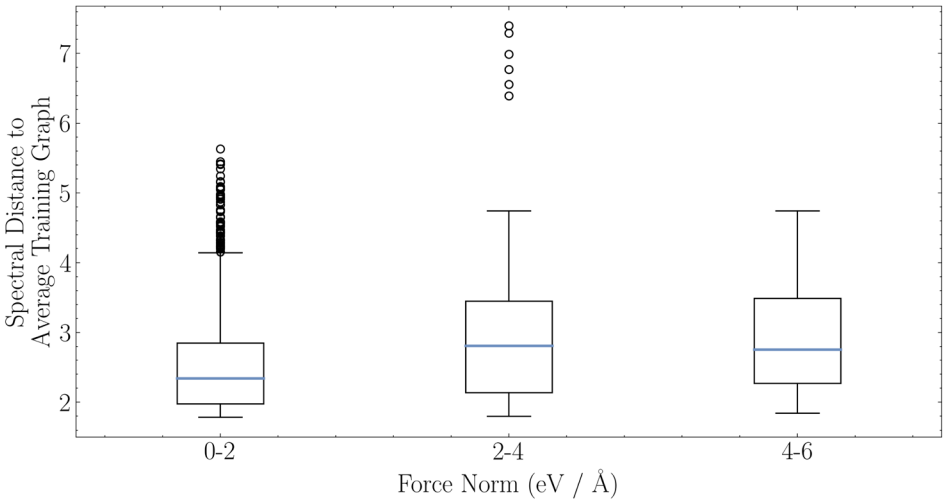


Fig. 20 Force norm *vs.* connectivity on SPICEv2. We analyze the force norms and connectivities of new molecules from the SPICEv2 dataset. The distribution of connectivities is similar across force different force norms. This implies that these distribution shifts can happen independently.

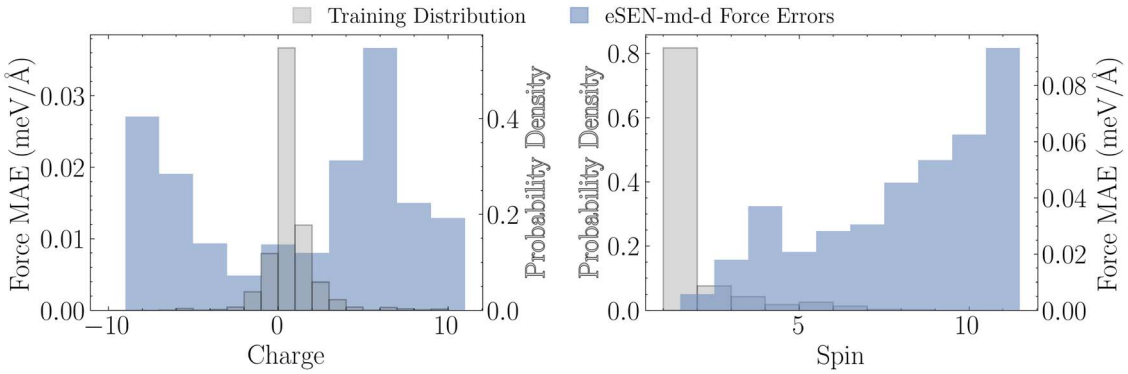


Fig. 21 Spin and charge distribution shift on OMol. We evaluate distribution shifts with respect to spin and charge. eSEN-md-d struggles with spins and charges poorly represented by the training data.[48] Note that since OMol contains very few highly charged structures, the average error in this regime is computed from limited samples, leading to noisier trends and wider variability in performance.

Acknowledgements

We thank Sanjeev Raja, Rasmus Lindrup, Yossi Gandelsman, Aayush Singh, Alyosha Efros, Eric Qu, and Yuan Chiang for the thoughtful discussions and feedback on this manuscript. This work was supported by the Toyota Research Institute as part of the Synthesis Advanced Research Challenge. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231.

References

1 I. Amin, S. Raja, and A. S. Krishnapriyan, Towards fast, specialized machine learning force fields: Distilling foundation models via energy hessians, in *The Thirteenth International Conference on Learning Representations*, 2025, https://openreview.net/forum?id=1durmugh3I.

2 N. Artrith, T. Morawietz and J. Behler, High-dimensional neural-network potentials for multicomponent systems: Applications to zinc oxide, *Phys. Rev. B: Condens. Matter Mater. Phys.*, 2011, **83**(15), 153101, DOI: 10.1103/physrevb.83.153101.

3 C. Bannwarth, S. Ehlert and S. Grimme, Gfn2-xtb—an accurate and broadly parametrized self-consistent tight-binding quantum chemical method with multipole electrostatics and density-dependent dispersion contributions, *J. Chem. Theory Comput.*, 2019, **15**(3), 1652–1671, DOI: 10.1021/acs.jctc.8b01176.

4 L. Barroso-Luque, M. Shuaibi, X. Fu, B. M. Wood, M. Dzamba, M. Gao, A. Rizvi, C. L. Zitnick, and Z. W. Ulissi, Open materials 2024 (omat24) inorganic materials dataset and models, *arXiv*, 2024, preprint, arXiv:2410.12771, DOI: 10.48550/arXiv.2410.12771, https://arxiv.org/abs/2410.12771.

5 I. Batatia, D. P. Kovacs, G. Simm, C. Ortner and G. Csányi, MACE: Higher order equivariant message passing neural

networks for fast and accurate force fields, *Adv. Neural Inf. Process. Syst.*, 2022, **35**, 11423–11436.

6 I. Batatia, P. Benner, Y. Chiang, A. M. Elena, D. P. Kovács, J. Riebesell, X. R. Advincula, M. Asta, M. Avaylon, W. J. Baldwin, F. Berger, N. Bernstein, A. Bhowmik, S. M. Blau, V. Cărare, J. P. Darby, S. De, F. D. Pia, V. L. Deringer, R. Elijošius, Z. El-Machachi, F. Falcioni, E. Fako, A. C. Ferrari, A. Genreith-Schriever, J. George, R. E. A. Goodall, C. P. Grey, P. Grigorev, S. Han, W. Handley, H. H. Heenen, K. Hermansson, C. Holm, J. Jaafar, S. Hofmann, K. S. Jakob, H. Jung, V. Kapil, A. D. Kaplan, N. Karimitari, J. R. Kermode, N. Kroupa, J. Kullgren, M. C. Kuner, D. Kuryla, G. Liepuoniute, J. T. Margraf, I.-B. Magdău, A. Michaelides, J. H. Moore, A. A. Naik, S. P. Niblett, S. W. Norwood, N. O'Neill, C. Ortner, K. A. Persson, K. Reuter, A. S. Rosen, L. L. Schaaf, C. Schran, B. X. Shi, E. Sivonxay, T. K. Stenczel, V. Svahn, C. Sutton, T. D. Swinburne, J. Tilly, C. van der Oord, E. Varga-Umbrich, T. Vegge, M. Vondrák, Y. Wang, W. C. Witt, F. Zills, and G. Csányi, A foundation model for atomistic materials chemistry, *arXiv*, 2024, preprint arXiv:2401.00096, DOI: 10.48550/arXiv.2401.00096.

7 S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt and B. Kozinsky, E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials, *Nat. Commun.*, 2022, **13**(1), 2453, DOI: 10.1038/s41467-022-29939-5.

8 M. Bechler-Speicher, I. Amos, R. Gilad-Bachrach, and A. Globerson, Graph neural networks use graphs when they shouldn't, *arXiv*, 2024, preprint, arXiv:2309.04332, DOI: 10.48550/arXiv.2309.04332, https://arxiv.org/abs/2309.04332.

9 J. Behler and M. Parrinello, Generalized neural-network representation of high-dimensional potential-energy surfaces, *Phys. Rev. Lett.*, 2007, **98**(14), 146401, DOI: 10.1103/physrevlett.98.146401.

10 V. Bihani, S. Mannan, U. Pratiush, T. Du, Z. Chen, S. Miret, M. Micoulaut, M. M. Smedskjaer, S. Ranu and N. M. A. Krishnan, Egraffbench: evaluation of equivariant graph neural network force fields for atomistic simulations, *Digital Discovery*, 2024, **3**(4), 759–768, DOI: 10.1039/d4dd00027g.

11 M. Bogojeski, L. Vogt-Maranto, M. E. Tuckerman, K.-R. Müller and K. Burke, Quantum chemical accuracy from density functional approximations via machine learning, *Nat. Commun.*, 2020, **11**(1), 5223, DOI: 10.1038/s41467-020-19093-1.

12 L. Chanussot, A. Das, S. Goyal, T. Lavril, M. Shuaibi, M. Riviere, K. Tran, J. Heras-Domingo, C. Ho, W. Hu, A. Palizhati, A. Sriram, B. Wood, J. Yoon, D. Parikh, C. L. Zitnick and Z. Ulissi, Open catalyst 2020 (oc20) dataset and community challenges, *ACS Catal.*, 2021, **11**(10), 6059–6072, DOI: 10.1021/acscatal.0c04525.

13 S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt and K.-R. Müller, Machine learning of accurate energy-conserving molecular force fields, *Sci. Adv.*, 2017, **3**(5), e1603015.

14 S. Chmiela, H. E. Sauceda, I. Poltavsky, K.-R. Müller and A. Tkatchenko, sgdml: Constructing accurate and data efficient molecular force fields using machine learning, *Comput. Phys. Commun.*, 2019, **240**, 38–45, DOI: 10.1016/j.cpc.2019.02.007.

15 S. Chmiela, V. Vassilev-Galindo, O. T. Unke, A. Kabylda, H. E. Sauceda, A. Tkatchenko and K.-R. Müller, Accurate global machine learning force fields for molecules with hundreds of atoms, *Sci. Adv.*, 2023, **9**(2), eadf0873.

16 F. Chung, *Spectral Graph Theory*, American Mathematical Society, 1996. ISBN 9781470424527, DOI: 10.1090/cbms/092.

17 D. R. Cox, The regression analysis of binary sequences, *J. R. Stat. Soc. B*, 1958, **20**(2), 215–232, DOI: 10.1111/j.2517-6161.1958.tb00292.x.

18 B. Deng, Y. Choi, P. Zhong, J. Riebesell, S. Anand, Z. Li, K. Jun, K. A. Persson, and G. Ceder, Overcoming systematic softening in universal machine learning interatomic potentials by fine-tuning, *arXiv*, 2024, preprint, arXiv:2405.07105, DOI: 10.48550/arXiv.2405.07105, https://arxiv.org/abs/2405.07105.

19 R. Drautz, Atomic cluster expansion for accurate and transferable interatomic potentials, *Phys. Rev. B*, 2019, **99**(1), 014104, DOI: 10.1103/PhysRevB.99.014104.

20 P. Eastman, P. K. Behara, D. L. Dotson, R. Galvelis, J. E. Herr, J. T. Horton, Y. Mao, J. D. Chodera, B. P. Pritchard, Y. Wang, G. De Fabritiis and T. E. Markland, Spice, a dataset of drug-like molecules and peptides for training machine learning potentials, *Sci. Data*, 2023, **10**(1), 11, DOI: 10.1038/s41597-022-01882-6.

21 P. Eastman, B. P. Pritchard, J. D. Chodera, and T. E. Markland Nutmeg and spice: Models and data for biomolecular machine learning, *arXiv*, 2024, preprint, arXiv:2406.13112, DOI: 10.48550/arXiv.2406.13112, https://arxiv.org/abs/2406.13112.

22 G. Fonseca, I. Poltavsky, V. Vassilev-Galindo and A. Tkatchenko, Improving molecular force fields across configurational space by combining supervised and unsupervised machine learning, *J. Chem. Phys.*, 2021, **154**(12), DOI: 10.1063/5.0035530.

23 A. I. Forrester, A. Sóbester and A. J. Keane, Multi-fidelity optimization via surrogate modelling, *Proc. R. Soc. A*, 2007, **463**(2088), 3251–3269, DOI: 10.1098/rspa.2007.1900.

24 X. Fu, Z. Wu, W. Wang, T. Xie, S. Keten, R. Gomez-Bombarelli and T. S. Jaakkola, Forces are not enough: Benchmark and critical evaluation for machine learning force fields with molecular simulations, *Transact. Mach. Learn. Res.*, 2023. ISSN 2835-8856. URL https://openreview.net/forum?id=A8pqQipwkt.

25 X. Fu, B. M. Wood, L. Barroso-Luque, D. S. Levine, M. Gao, M. Dzamba, and C. L. Zitnick, Learning smooth and expressive interatomic potentials for physical property prediction, *arXiv*, 2025, preprint, arXiv:2502.12147, DOI: 10.48550/arXiv.2502.12147, https://arxiv.org/abs/2502.12147.

26 P. Fuchs, S. Thaler, S. Röcken and J. Zavadlav, chemtrain: Learning deep potential models via automatic differentiation and statistical physics, *Comput. Phys.*

Commun., 2025, **310**, 109512, DOI: 10.1016/j.cpc.2025.109512.

27 Y. Gandelsman, Y. Sun, X. Chen and A. A. Efros, Test-time training with masked autoencoders, *Adv. Neural Inf. Process. Syst.*, 2022, **35**, 29374–29385.

28 J. L. A. Gardner, K. T. Baker and V. L. Deringer, Synthetic pre-training for neural-network interatomic potentials, *Mach. Learn.: Sci. Technol.*, 2024, **5**(1), 015003, DOI: 10.1088/2632-2153/ad1626.

29 J. Gasteiger, F. Becker and S. Günnemann, Gemnet: Universal directional graph neural networks for molecules, *Adv. Neural Inf. Process. Syst.*, 2021, **34**, 6790–6802.

30 J. Gasteiger, M. Shuaibi, A. Sriram, S. Günnemann, Z. Ulissi, C. L. Zitnick, and A. Das, Gemnet-oc: Developing graph neural networks for large and diverse molecular simulation datasets, *arXiv*, 2022, preprint, arXiv:2204.02782, DOI: 10.48550/arXiv.2204.02782.

31 J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, Neural message passing for quantum chemistry, *International Conference on Machine Learning*, 2017.

32 F. D. Giovanni, L. Giusti, F. Barbero, G. Luise, P. Lio', and M. Bronstein, On over-squashing in message passing neural networks: The impact of width, depth, and topology, *arXiv*, 2023, preprint, arXiv:2302.02941, DOI: 10.48550/arXiv.2302.02941, https://arxiv.org/abs/2302.02941.

33 M. Giselle Fernández-Godino, Review of multi-fidelity models, *Adv. Comput. Sci. Eng.*, 2023, **1**(4), 351–400, DOI: 10.3934/acse.2023015.

34 S. Gunasekar, J. Lee, D. Soudry, and N. Srebro, Implicit bias of gradient descent on linear convolutional networks, *arXiv*, 2019, preprint, arXiv:1806.00468, DOI: 10.48550/arXiv.1806.00468, https://arxiv.org/abs/1806.00468.

35 B. Han and K. Yu, Refining potential energy surface through dynamical properties via differentiable molecular simulation, *Nat. Commun.*, 2025, **16**, 816, DOI: 10.1038/s41467-025-56061-z.

36 M. Hardt and Y. Sun, Test-time training on nearest neighbors for large language models, *The Twelfth International Conference on Learning Representations*, 2024.

37 S. Heinen, D. Khan, G. F. von Rudorff, K. Karandashev, D. J. Arismendi Arrieta, A. Price, S. Nandi, A. Bhowmik, K. Hermansson and A. von Lilienfeld, Reducing training data needs with minimal multilevel machine learning (M3L), *Mach. Learn.: Sci. Technol.*, 2024, **5**, 025058.

38 D. Hendrycks and T. Dietterich, Benchmarking neural network robustness to common corruptions and perturbations, *arXiv*, 2019, preprint, arXiv:1903.12261, DOI: 10.48550/arXiv.1903.12261, https://arxiv.org/abs/1903.12261.

39 A. Hjorth Larsen, J. Jørgen Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. Bjerre Jensen, J. Kermode, J. R. Kitchin, E. Leonhard Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. Bergmann Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng and K. W. Jacobsen, The atomic simulation environment—a python library for working with atoms, *J. Phys.: Condens. Matter*, 2017, **29**(27), 273002, DOI: 10.1088/1361-648X/aa680e.

40 D. G. Horvitz and D. J. Thompson, A generalization of sampling without replacement from a finite universe, *J. Am. Stat. Assoc.*, 1952, **47**(260), 663–685, DOI: 10.1080/01621459.1952.10483446.

41 K. Jacobsen, P. Stoltze and J. Nørskov, A semi-empirical effective medium theory for metals and alloys, *Surf. Sci.*, 1996, **366**(2), 394–402, DOI: 10.1016/0039-6028(96)00816-3. https://www.sciencedirect.com/science/article/pii/0039602896008163.

42 M. Jang, S.-Y. Chung, and H. W. Chung Test-time adaptation via self-training with nearest neighbor information, *arXiv*, 2023, preprint, arXiv:2207.10792, DOI: 10.48550/arXiv.2207.10792.

43 T. Jeong and H. Kim, *Ood-maml: Meta-learning for few-shot out-of-distribution detection and classification*, Curran Associates, Inc., 2020, volume 33, pp. 3907–3916, https://proceedings.neurips.cc/paper_files/paper/2020/file/28e209b61a52482a0ae1cb9f5959c792-Paper.pdf.

44 D. Jha, K. Choudhary, F. Tavazza, W.-K. Liao, A. Choudhary, C. Campbell and A. Agrawal, Enhancing materials property prediction by leveraging computational and experimental data using deep transfer learning, *Nat. Commun.*, 2019, **10**(1), 5316, DOI: 10.1038/s41467-019-13297-w.

45 I. Jovanović and Z. Stanić, Spectral distances of graphs, *Linear Algebra Appl.*, 2012, **436**(5), 1425–1435, DOI: 10.1016/j.laa.2011.08.019.

46 D. P. Kovács, J. H. Moore, N. J. Browning, I. Batatia, J. T. Horton, V. Kapil, W. C. Witt, I.-B. Magdău, D. J. Cole and G. Csányi, Mace-off23: Transferable machine learning force fields for organic molecules, *J. Am. Ceram. Soc.*, 2025, **141**(21), 17598–17611.

47 M. Kulichenko, B. Nebgen, N. Lubbers, J. S. Smith, K. Barros, A. E. A. Allen, A. Habib, E. Shinkle, N. Fedik, Y. W. Li, R. A. Messerly and S. Tretiak, Data generation for machine learning interatomic potentials and beyond, *Chem. Rev.*, 2024, **124**(24), 13681–13714, DOI: 10.1021/acs.chemrev.4c00572.

48 D. S. Levine, M. Shuaibi, E. W. C. Spotte-Smith, M. G. Taylor, M. R. Hasyim, K. Michel, I. Batatia, G. Csányi, M. Dzamba, P. Eastman, N. C. Frey, X. Fu, V. Gharakhanyan, A. S. Krishnapriyan, J. A. Rackers, S. Raja, A. Rizvi, A. S. Rosen, Z. Ulissi, S. Vargas, C. L. Zitnick, S. M. Blau, and B. M. Wood The open molecules 2025 (omol25) dataset, evaluations, and models, *arXiv*, 2025, preprint, arXiv:2505.08762, DOI: 10.48550/arXiv.2505.08762, https://arxiv.org/abs/2505.08762.

49 K. Li, A. N. Rubungo, X. Lei, D. Persaud, K. Choudhary, B. DeCost, A. B. Dieng and J. Hattrick-Simpers, Probing out-of-distribution generalization in machine learning for materials, *Commun. Mater.*, 2025, **6**(1), 9, DOI: 10.1038/s43246-024-00731-w.

50 Y.-L. Liao, B. Wood, A. Das, and T. Smidt Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations, *arXiv*, 2024, preprint, arXiv:2306.12059, DOI: 10.48550/arXiv.2306.12059, https://arxiv.org/abs/2306.12059.

51 J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence *When Training and Test Sets Are Different: Characterizing Learning Transfer*, 2009, pp. 3–28.

52 S. Raja, I. Amin, F. Pedregosa and A. S. Krishnapriyan, Stability-aware training of machine learning force fields with differentiable boltzmann estimators, *Transact. Mach. Learn. Res.*, 2025. https://openreview.net/forum?id=ZckLMG00sO.

53 R. Ramakrishnan, P. O. Dral, M. Rupp and O. A. von Lilienfeld, Big data meets quantum chemistry approximations: The Δ-machine learning approach, *J. Chem. Theory Comput.*, 2015, **11**(5), 2087–2096, DOI: 10.1021/acs.jctc.5b00099.

54 B. Recht, R. Roelofs, L. Schmidt, and V. Shankar, Do cifar-10 classifiers generalize to cifar-10?, *arXiv*, 2018, preprint, arXiv:1806.00451, DOI: 10.48550/arXiv.1806.00451, https://arxiv.org/abs/1806.00451.

55 M. Schreiner, A. Bhowmik, T. Vegge, J. Busk, and O. Winther, Transition1x – a dataset for building generalizable reactive machine learning potentials, *arXiv*, 2022, preprint, arXiv:2207.12858, DOI: 10.48550/arXiv.2207.12858, https://arxiv.org/abs/2207.12858.

56 P. Schwerdtfeger and D. J. Wales, 100 years of the lennard-jones potential, *J. Chem. Theory Comput.*, 2024, **29**(9), 3379–3405, DOI: 10.1021/acs.jctc.4c00135.

57 K. Schütt, P.-J. Kindermans, H. E. Sauceda Felix, S. Chmiela, A. Tkatchenko and K.-R. Müller, Schnet: A continuous-filter convolutional neural network for modeling quantum interactions, *Adv. Neural Inf. Process. Syst.*, 2017, **30**.

58 N. Shoghi, A. Kolluru, J. R. Kitchin, Z. W. Ulissi, C. L. Zitnick, and B. M. Wood, From molecules to materials: Pre-training large generalizable models for atomic property prediction, *arXiv*, 2023, preprint, arXiv:2310.16802, DOI: 10.48550/arXiv.2310.16802.

59 Z. Shui, D. S. Karls, M. Wen, I. A. Nikiforov, E. B. Tadmor and G. Karypis, Injecting domain knowledge from empirical interatomic potentials to neural networks for predicting material properties, *Adv. Neural Inf. Process. Syst.*, 2022, 14839–14851.

60 J. S. Smith, R. Zubatyuk, B. Nebgen, N. Lubbers, K. Barros, A. E. Roitberg, O. Isayev and S. Tretiak, The ani-1ccx and ani-1x data sets, coupled-cluster and density functional theory properties for molecules, *Sci. Data*, 2020, 7(1), 134, DOI: 10.1038/s41597-020-0473-z.

61 M. Sugiyama, M. Krauledat and K.-R. Müller, Covariate shift adaptation by importance weighted cross validation, *J. Mach. Learn. Res.*, 2007, **8**(35), 985–1005. http://jmlr.org/papers/v8/sugiyama07a.html.

62 Y. Sun, X. Wang, Z. Liu, J. Miller, A. A. Efros, and M. Hardt, Test-time training with self-supervision for generalization under distribution shifts, *International Conference on Machine Learning*, 2020.

63 R. Taori, A. Dave, V. Shankar, N. Carlini, B. Recht and L. Schmidt, Measuring robustness to natural distribution shifts in image classification, *Adv. Neural Inf. Process. Syst.*, 2020, **33**, 18583–18599.

64 R. Tran, J. Lan, M. Shuaibi, B. M. Wood, S. Goyal, A. Das, J. Heras-Domingo, A. Kolluru, A. Rizvi, N. Shoghi, A. Sriram, F. Therrien, J. Abed, O. Voznyy, E. H. Sargent, Z. Ulissi and C. L. Zitnick, The open catalyst 2022 (oc22) dataset and challenges for oxide electrocatalysts, *ACS Catal.*, 2023, **13**(5), 3066–3084, DOI: 10.1021/acscatal.2c05426.

65 J. Vandermause, S. B. Torrisi, S. Batzner, Y. Xie, L. Sun, A. M. Kolpak and B. Kozinsky, On-the-fly active learning of interpretable bayesian force fields for atomistic rare events, *npj Comput. Mater.*, 2020, **6**(1), 20, DOI: 10.1038/s41524-020-0283-z.

66 V. Vinod, S. Maity, P. Zaspel and U. Kleinekathöfer, Multifidelity machine learning for molecular excitation energies, *J. Chem. Theory Comput.*, 2023, **19**(21), 7658–7670, DOI: 10.1021/acs.jctc.3c00882.

67 R. C. Wilson and P. Zhu, A study of graph spectra for comparing graphs and trees, *Pattern Recognit.*, 2008, **41**(9), 2833–2841, DOI: 10.1016/j.patcog.2008.03.011. https://www.sciencedirect.com/science/article/pii/S0031320308000927.

68 S. Zaidi, M. Schaarschmidt, J. Martens, H. Kim, Y. W. Teh, A. Sanchez-Gonzalez, P. Battaglia, R. Pascanu, and J. Godwin, Pre-training via denoising for molecular property prediction. *International Conference on Learning Representations*, 2022.

69 L. Zhang, J. Han, H. Wang, R. Car and W. E, Deep potential molecular dynamics: A scalable model with the accuracy of quantum mechanics, *Phys. Rev. Lett.*, 2018, **120**(14), 143001, DOI: 10.1103/physrevlett.120.143001.

70 Y. Zhang, S. Nie, W. Liu, X. Xu, D. Zhang and H. T. Shen, Sequence-to-sequence domain adaptation network for robust text image recognition, *Conference on Computer Vision and Pattern Recognition*, 2019, pp. 2735–2744, DOI: 10.1109/CVPR.2019.00285.

71 B. Zhao, S. Yu, W. Ma, M. Yu, S. Mei, A. Wang, J. He, A. Yuille, and A. Kortylewski, *Ood-cv: A benchmark for robustness to out-of-distribution shifts of individual nuisances in natural images*, Springer, 2022, pp. 163–180.

72 K. Zhou, Y. Yang, Y. Qiao and T. Xiang, Domain adaptive ensemble learning, *IEEE Trans. Image Process.*, 2021, **30**, 8008–8018, DOI: 10.1109/TIP.2021.3112012.

73 T. Kreiman, Y. Bai, F. Atieh, E. Weaver, E. Qu and A. Krishnapriyan, Transformers Discover Molecular Structure Without Graph Priors, *arXiv*, 2025, preprint, arXiv:2510.02259, DOI: 10.48550/arXiv.2510.02259, https://arxiv.org/abs/2510.02259.