Data efficiency and extrapolation trends in neural network interatomic potentials

IOP Publishing Mach. Learn.: Sci. Technol. 4 (2023) 035031 https://doi.org/10.1088/2632-2153/acf115

MACHINE LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
5 May 2023

REVISED
21 July 2023

ACCEPTED FOR PUBLICATION
16 August 2023

PUBLISHED
25 August 2023

PAPER

Data efficiency and extrapolation trends in neural network interatomic potentials

Joshua A Vita[1,2] and Daniel Schwalbe-Koda[2,*]

[1] Lawrence Livermore National Laboratory, Livermore, CA 94550, United States of America
[2] Department of Materials Science and Engineering, University of Illinois at Urbana-ChampaignUniversity of Illinois at Urbana-Champaign, Urbana, IL 61801, United States of America
[*] Author to whom any correspondence should be addressed.

E-mail: dskoda@llnl.gov

Keywords: neural network potentials, extrapolation, loss landscapes, graph neural networks, machine learning potentials, atomistic simulations

Supplementary material for this article is available online

Abstract

Recently, key architectural advances have been proposed for neural network interatomic potentials (NNIPs), such as incorporating message-passing networks, equivariance, or many-body expansion terms. Although modern NNIP models exhibit small differences in test accuracy, this metric is still considered the main target when developing new NNIP architectures. In this work, we show how architectural and optimization choices influence the generalization of NNIPs, revealing trends in molecular dynamics (MD) stability, data efficiency, and loss landscapes. Using the 3BPA dataset, we uncover trends in NNIP errors and robustness to noise, showing these metrics are insufficient to predict MD stability in the high-accuracy regime. With a large-scale study on NequIP, MACE, and their optimizers, we show that our metric of loss entropy predicts out-of-distribution error and data efficiency despite being computed only on the training set. This work provides a deep learning justification for probing extrapolation and can inform the development of next-generation NNIPs.

1. Introduction

Machine learning has proven extremely valuable in the materials and chemical sciences as a tool for data analysis and generation [1–4]. Particularly in atomistic simulations, ML-based models offer a compelling balance between high-accuracy, high-cost quantum chemistry calculations and low-accuracy, low-cost classical force fields [5–7]. Whereas several models based on kernel regression or Gaussian processes have been proposed [6, 8–12], developments in neural network (NN) interatomic potentials (IPs) have shown promise due to their low inference time, scalability to large datasets, and high accuracy in predicting potential energy surfaces (PESes) [7, 9, 13]. These methods have been used for a variety of applications, including molecular simulation, excited-state dynamics, phase transitions, chemical reactions, and more [7, 9, 14–16].

Over the last few years, several different model architectures were proposed to reduce errors in PES fitting and decrease the amount of data required to train the models. For example, neural network interatomic potentials (NNIPs) were first proposed using feed-forward NNs and symmetry-based representations [13], but have since been improved by designing new representations to better capture the atomic environment [17–22]. Most recently, message-passing NNs (MPNNs) [23] have shown remarkable ability to fit PESes using learned representations. In particular, NN architectures incorporating physics concepts such as directional representations and equivariance [24–27] or many-body interactions [28–30] have gained popularity due to higher accuracy and data efficiency. Despite their successes, NNIPs still struggle with data efficiency and robust generalization. Recent works show that accuracy metrics over datasets are insufficient to quantify the models' quality in production simulations and motivate the use of alternative metrics such as computational speed or simulation stability [31–36]. Furthermore, different NNIP models may have similar test accuracy, but completely different extrapolation ability [37, 38]. This begs the question: **which metrics can distinguish between NNIPs with similar test error but different extrapolation behavior?**

In this work, we uncover trends in NNIP robustness and show how loss landscapes (LLs) of the training set predict extrapolation power in NNIPs. In analogy with other deep learning applications [39–42] and visualizations of the LLs [43–46], we propose a metric of loss entropy for NNIPs without the cost associated to the Hessian of the loss [47, 48]. In particular, we provide the following contributions:

- Using literature data and two state-of-the-art NNIP models (NequIP and MACE), we show that extrapolation trends can be obtained from error metrics in the 3BPA dataset. For example, the NNIPs are able to recover the underlying PES despite being trained to noisy labels, and have extrapolation errors that follow scaling relations against in-domain accuracy metrics. This scaling relation demonstrates the challenges associated with decoupling the effects of implicit data regularization from those of model architecture changes and emphasizes the need for alternative metrics for quantifying model performance.
- To circumvent the limitations above, we show that LLs can provide evaluation strategies for NNIPs beyond accuracy metrics. Qualitative inspection of LLs explains some heuristic training regimes for NNIPs, such as the use of higher weights for forces loss, weight cycling (WC), or separating learning rates for different parts of the architecture, thus providing theoretical justification for certain hyperparameters.
- Using a large-scale study with NequIP and MACE, we show that flatness of LLs correlates with model robustness. In particular, we quantify the loss entropy around the optimized models and relate them to errors in the extrapolation regime. Furthermore, we show that molecular dynamics (MD) simulations of models with flatter LLs exhibit less unphysical behavior than their sharper counterparts.

This approach can guide the development of newer NNIP architectures with higher accuracy, trainability, and robustness.

2. Methods

2.1. Visualizing loss landscapes

The loss landscape ℓ of a NN can be visualized by evaluating the loss function $\mathcal{L}$ along a trajectory between two parameter sets $\boldsymbol{\theta}$ and $\boldsymbol{\theta}'$. The simplest approach is to linearly interpolate the weights [43], choosing a scalar $t \in [0, 1]$ such that $\boldsymbol{\theta}(t) = (1-t)\boldsymbol{\theta} + t\boldsymbol{\theta}'$. Then, the loss landscape ℓ for a model becomes

$$\ell(t) = \mathcal{L}\left(\boldsymbol{\theta}(t)\right) = \mathcal{L}\left((1-t)\boldsymbol{\theta} + t\boldsymbol{\theta}'\right). \tag{1}$$

In this work, the loss $\mathcal{L}$ is evaluated on the training set of the model with parameters $\boldsymbol{\theta}$ on a given dataset.

In the absence of the reference weights $\boldsymbol{\theta}'$, the LL can be constructed by sampling a random vector $\boldsymbol{\delta}$ in the parameter space and plotting the LL around $\boldsymbol{\theta}$ as

$$\ell(t; \boldsymbol{\delta}) = \mathcal{L}\left(\boldsymbol{\theta} + t\boldsymbol{\delta}\right), \tag{2}$$

where the domain of t is appropriately chosen to span a neighborhood of $\boldsymbol{\theta}$. This notion can be extended to 2D LLs by taking two orthogonal vectors $\boldsymbol{\delta}_1, \boldsymbol{\delta}_2$ such that

$$\ell(t_1, t_2; \boldsymbol{\delta}_1, \boldsymbol{\delta}_2) = \mathcal{L}\left(\boldsymbol{\theta} + t_1\boldsymbol{\delta}_1 + t_2\boldsymbol{\delta}_2\right), \tag{3}$$

with scalars t_1, t_2 chosen to span a (two-dimensional) neighborhood of $\boldsymbol{\theta}$. These approaches have been used to study the LLs of NN classifiers in image datasets, interpolate between sets of classifiers, and explore the loss function around degenerate minima [47, 49–51].

One challenge when analyzing LLs is comparing different models according to their parameters. Because activation functions such as ReLU allow for scale-invariance of NN weights, especially when coupled with batch normalization techniques, the magnitude of the vector $\boldsymbol{\delta}$ is not transferable from model to model. This prevents a fair comparison of LLs, curvatures, and sharpness metrics. To account for this, we use the filter normalization technique proposed by Li *et al* [44]. Therein, each random vector $\boldsymbol{\delta}$ is normalized by the scale of each filter i, in each layer j of the NN, i.e.

$$\bar{\boldsymbol{\delta}}^{(i,j)} = \frac{\boldsymbol{\delta}^{(i,j)}}{\|\boldsymbol{\delta}^{(i,j)}\|}\|\boldsymbol{\theta}^{(i,j)}\|, \tag{4}$$

where $\|.\|$ is the Frobenius norm. Then, the LL is plotted according to the filter-normalized vector $\bar{\boldsymbol{\delta}}$,

$$\ell(t; \bar{\boldsymbol{\delta}}) = \mathcal{L}\left(\boldsymbol{\theta} + t\bar{\boldsymbol{\delta}}\right), \tag{5}$$

and analogously for 2D LLs.

Although informative, sampling LLs can have cost comparable to training the NN, since evaluating the loss for each interpolated weight in each direction is similar to one training epoch. Depending on the number of parameters and directions $\boldsymbol{\delta}_n$ under analysis, the loss may be evaluated over the entire training dataset a large number of times.

2.2. Quantitative comparison of loss landscapes

Despite the usefulness of visualizing loss landscapes, comparing them beyond qualitative insights requires figures of merit to differentiate them. The most commonly used metric in the field is the curvature of the LL [42], which is related to the magnitude of the eigenvalues of the loss Hessian. Although informative, the Hessian is a local property and cannot fully capture 'valley-like' degeneracies in loss landscapes. Furthermore, computing the full Hessian for a system with millions of training parameters would be intractable. Thus, alternative metrics to compare LLs are required.

In the case of NNIPs, which are often trained to both energies and forces, two LLs are obtained per model, one for each target value. Although they cannot be completely disentangled, both LLs should be compared simultaneously to assess model performance. To derive one of such metrics, we propose the use of 'loss entropy' [41, 52] to quantify loss flatness around optimized minima. Although variations of this quantity have been proposed, we use the formula

$$S(T) = k\log\left[\sum_t \exp\left(\frac{-\bar{\ell}(t)}{kT}\right)\right], \tag{6}$$

where S is the entropy of the loss landscape $\bar{\ell}(t)$ computed with respect to energy or forces and averaged over N orthogonal, filter-normalized weight displacements,

$$\bar{\ell}(t) = \frac{1}{N}\sum_n \ell(t;\bar{\boldsymbol{\delta}}_n) \tag{7}$$

and kT is a weighting parameter that quantifies the 'flatness' of the LL with respect to a certain acceptable threshold of training loss. This is similar to distributions of microstates accessible at a given 'temperature,' and ensures that low-loss states contribute to a much larger entropy than high-loss states.

As the entropy of the LL of a NNIP takes into account both energy and forces losses, we adopt a strategy similar to the one used during training of the NNs by balancing energy and forces losses with a weighted sum,

$$S = \alpha S_E(T_E) + (1-\alpha)S_F(T_F), \tag{8}$$

where α is a dimensionless parameter between 0 and 1 that weights the entropy of the energy loss (S_E) and forces loss (S_F). As these losses have different units, they have to be computed with different normalization parameters $k_E T_E$ and $k_F T_F$, each of which takes into account 'thermal randomness' of the training loss. For simplicity, in this work, we adopt $k = 1$ as a dimensionless parameter, and assign the adequate units to the 'thermal error' for better interpretability of the loss entropy.

2.3. NNIPs and dataset

NequIP [27] is an equivariant NNIP that uses Clebsch–Gordon transformations and spherical harmonics to incorporate equivariance in the model. NequIP demonstrates state-of-the-art accuracy in several datasets, data efficiency, and has been employed to simulate a variety of organic and inorganic systems.

MACE [29] is an equivariant NNIP that uses higher-order messages to efficiently embed information beyond two-body interactions in traditional MPNNs. The model demonstrates state-of-the-art performance in a variety of benchmarks, faster learning, and competitive computational cost.

Other NNIPs proposed recently and not benchmarked in this study include Allegro [30], GemNet [53], DimeNet [54], HIP-NN [55], NewtonNet [56], BOTNet [28], SchNet [57], PaiNN [58], and others [18, 20, 26, 59].

Training details are provided in appendix A in the supporting information.

The 3BPA dataset [32] under study in this work was chosen due to its previous use in the literature for benchmarking NNIP models in extrapolation behavior [28–30]. The benchmark involves training models on low-temperature samples and evaluating their performance on held-out samples from high-temperature simulations. Distributions of energies and forces of the 3BPA dataset are shown in D.1 in the supporting information, figure S2. Additional results using the ANI-Al dataset [60] be found in appendix D.2 in the supporting information.

2.4. MD simulations

Molecular dynamics simulations were performed for the 3BPA molecule using each of the models under study. In total, 30 trajectories were performed per model to obtain better statistics of their production behavior. Simulations were performed in the NVT ensemble using the Berendsen thermostat [61] implemented in ASE [62]. The initial configuration used for the simulation was chosen to be the ground state of the 3BPA molecule. The simulation was performed at 1600 K and a timestep of 1 fs to force the models to evaluate configurations outside of their training set (300 K samples) and beyond those quantified by the test set errors (1200 K samples). The time constant of the Berendsen thermostat was chosen to be 250 fs. The simulation was performed for 6 ps for all models. This time length was sufficient to ensure the temperature was equilibrated and remained at 1600 K for about 4 ps (see figure S8).

MD trajectories were considered unphysical if the distance between bonded atoms increased above 2 Å. This bond rupture was the main source of failure observed in the trajectories, and thus is the only one considered in this work.

3. Results

As is discussed in the Introduction, the central theme of this work is to explore the suitability of different metrics for predicting the extrapolation capacity of NNIPs. We approach this problem by considering two metrics— root mean squared error (RMSE) and loss entropy—and assessing their ability to predict two metrics quantifying the extrapolation capacity of a model: the time to failure of MD simulations and the slope of the learning curve [27, 28]. In the first part, we observe that NNIPs are capable of learning smoothed PESes from noisy data—an extrapolation power that may be a consequence of implicit regularization from the dataset—further motivating the need of an alternative performance metric. We then demonstrate how loss landscapes can provide qualitative understanding of model training behaviors, and how loss entropy correlates well with time to failure for NequIP and MACE models. Finally, we further confirm the ability of the loss entropy to predict out-of-domain behavior for both models, as quantified using the slope of the learning curve.

3.1. Trends in robustness to noise of NNIPs

When comparing NNIPs, metrics of interest typically include errors in predicting forces and energies of a test dataset, and are appropriately used as baselines for assessing model quality. However, accuracy metrics are not necessarily predictive of extrapolation power [33]. A simple test to consider when analyzing NNIP extrapolation is whether NNIPs can learn a PES despite being trained on noisy data. Although this test does not measure extrapolation to out-of-distribution data, it verifies whether models are expected to overfit to corrupted training data, which would lower their robust generalization power. For example, NN-based classifiers can overfit to random labels in image datasets or to completely random inputs [63], even in architectures with good generalization error that are designed to prevent overfitting. Most NNIPs have enough parameters to memorize the training data, but standard regularization and architectural choices can curb overfitting in NNIPs, leading to lower generalization errors.

To test this hypothesis, we trained four different NNIPs to the 3BPA dataset and analyzed their *training* error. Following the lead from the deep learning literature [63], we then gradually corrupted the labels of the training set by adding a random sample from $\mathcal{N}(0, \sigma \cdot \sigma_{\mathrm{DFT}})$ to the true forces, where σ_{DFT} is the standard deviation in the distribution of forces for the dataset obtained with density functional theory (DFT) (see appendix D.1 in the supporting information), and σ is a scalar ranging from 0.0 to 0.1. In principle, NN regressors with arbitrary levels of expressivity (or absent regularization) could achieve low training error even in these noisy PESes. Figure 1(a) shows the error for NNIPs trained to the 3BPA dataset with corrupted forces, and tested on the original, un-corrupted data. When the forces are not corrupted, models exhibit reasonable training errors lower than 40 meV $\mathrm{Å}^{-1}$, as expected by their nominal performances in energy prediction [27, 29]. However, even small amounts of noise in the forces prevent the noisy dataset from being memorized with high accuracy, with the training loss plateauing instead of tending to zero. The ability of the models to predict the noisy forces saturates at the limit of the noise, indicating that these NNIPs do not memorize the high-frequency labels. On the other hand, when the test error of the models trained with corrupted labels is computed with respect to the uncorrupted dataset, the error is substantially smaller than the noise baseline (see the 'original' panel of figure 1(a)). Thus, the NNIPs under analysis are able to learn the underlying PES in the 3BPA dataset despite the added noise.

Contrary to the overfitting hypothesis, these results suggest that data regularization in the training set may help NNIPs to 'denoise' the data. To understand this effect, we demonstrate in appendix B how data redundancy can downplay the effect of external noise in a toy system. Figure S1 shows how a large number of training points can counterbalance the effect of external noise when predicting the original, non-noisy data

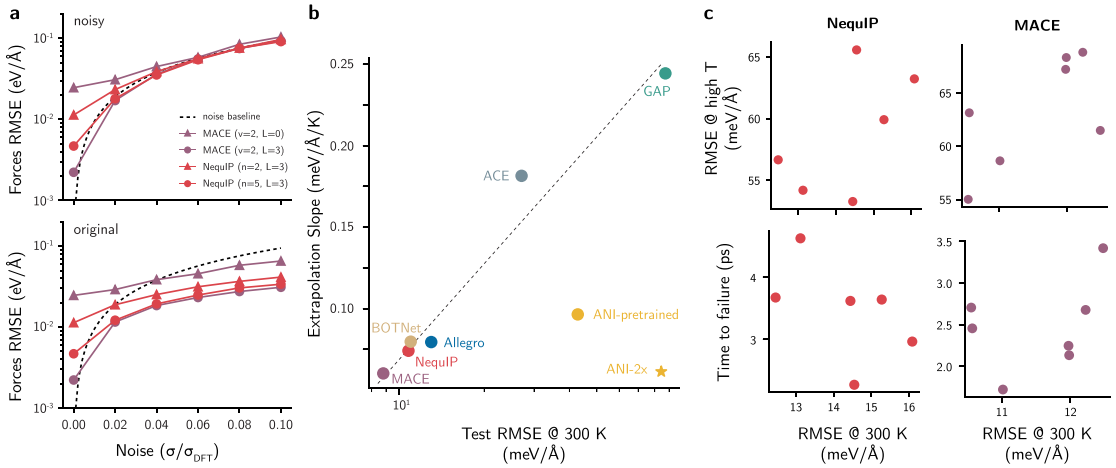

Figure 1. (a) Forces RMSE values (eV $Å^{-1}$) for models trained to only the forces of increasingly noisy versions of the 3BPA dataset. The models trained on noisy data are then evaluated on their noisy training set ('noisy' panel) or on the original, uncorrupted dataset ('original'). The dashed lines correspond to the amount of noise that was added to the DFT forces in units of eV $Å^{-1}$. The legend denotes the maximum body order of the MACE models (ν), the number of interaction layers in the NequIP models (n), or the maximum order of the model irreps (L). All MACE models in (a) used only two interaction layers. (b) Relationship between extrapolation ability, quantified using the slope of the test errors on the 3BPA dataset with increasing temperature ('extrapolation slope'), and the test error at 300 K. Slopes were computed by performing a linear fit to values taken from the literature for MACE [29], BOTNet [28], and others [30] (see figure S3). The dashed line is a visual guide. (c) Forces RMSE on the 300 K test set of several NequIP and MACE models with different architectures, but similar testing errors on the 300 K split. (top) Correlation between test errors on all splits (300, 600, and 1200 K) and test error on the 300 K split for models with similar test errors. (bottom) Average time-to-failure of MD simulation for each of the models with similar test errors.

for the case of linear regressor models. In this case, the model averages out the noise and recovers the true function even at high levels of added noise. On the other hand, at the low-data regime, the regression model is unable to recover the true function, and its error quickly grows. Although the results from the linear model may not directly translate to the case of NNs, figure 1(a) shows that, to an extent, NNIPs are able to 'denoise' energies from the dataset due to architectural, training, or implicit data regularization.

To verify if this behavior was specific to the 3BPA dataset or could generalize to other datasets, we corrupted the energies or forces of the ANI-Al dataset and trained different models on the noisy values (see appendix D.2 in the supporting information), obtaining similar results even when only energies were considered. As the ANI-Al example exhibited even better results than the 3BPA, we tested whether a drastic increase in the injected noise could still be denoised by the NNIPs under study. Following the results of figure S1, we trained the two NNIP architectures under study to PESes with energy noises up to twenty times higher than those in figure 1(see figure S12), and up to twice the standard deviation of the original dataset distribution of the ANI-Al dataset. Although the distribution of per-atom energies shows that the noisy PES is completely different than the original one (figures S12(b) and (c)), all models succeeded in modeling the underlying PES below the error baseline (figure S12). As also illustrated by the toy example, the performance of the models degrades as extremely large amounts of noise are added. Nonetheless, errors with respect to the non-noisy dataset are remarkably low considering the corruption baselines.

The toy example from appendix B in the supporting information can be considered an upper bound of the 'dataset denoising' ability, given that a functional form of the inputs is known and the model could, in principle, fit perfectly to the data. As the trends of NNIPs trained on the 3BPA or ANI-Al potential approach this behavior, it can be concluded that the two NNIP architectures are able to 'denoise' the external noise added to the datasets, possibly due to data redundancy. As generalization tests assume the model is being tested on unseen data, it is not clear whether the accuracy reflects the quality of the model predictions or simply their ability to reproduce local environments existing in the training data. This is particularly important in the high-data regime, when the test dataset may be correlated to the train dataset in non-obvious ways.

3.2. Trends in extrapolation power of NNIPs

To bypass the data correlation problem, alternative strategies were proposed to measure generalization power, including separating train-test splits according to sampling temperature, as is the case of the 3BPA dataset [32]. While testing a model on held-out samples from high-temperature simulations is typically considered an independent evaluation of its performance, we have found that extrapolation errors are correlated with low-temperature test errors across various model architectures. This is performed by fitting a

linear model to the errors on the 3BPA testing sets at 300, 600, and 1200 K from the literature [28–30] (see figure S3), then using the slope of the fitted line as the associated metric. Figure 1(b) shows that all the models that were trained only to 3BPA frames at 300 K follow an approximate linear scaling relation between the extrapolation slope and the log of the low-temperature errors. This correlation between the low- and high-temperature data does not strictly preclude the use of the 3BPA dataset for assessing a fitted model's extrapolation abilities, as the extrapolation slope represents how much the accuracy of a given model degrades as the sampling temperature increases. Nevertheless, it suggests that data and model regularization effects may be enforcing extrapolation trends in wide error ranges. For example, a model like ACE [22, 28] is known to provide functional forms that aid extrapolation beyond the training data [32]. Despite this, the ACE model for 3BPA follows the same trends of more complex message-passing NNs. In fact, the generalization slopes are correlated to their low-temperature error regardless of significant architectural differences. This indicates that, for this benchmark, the RMSEs at higher temperatures can be estimated from the test error at 300 K even without evaluating the model at the other test sets.

The exceptions to this rule are the two ANI models ('ANI-2x' and 'ANI-pretrained'), which were pre-trained to the 8.9 million configurations from the ANI-2x dataset [64]. As seen in other fields of deep learning [65], the pre-trained models extrapolate better than all other models, though fine-tuning on 3BPA ('ANI-pretrained') leads to slightly worse extrapolation slope. These results suggest that: (1) more diverse datasets may be required for assessing the extrapolation and generalization capacity of a model; and (2) pre-training on large datasets may be required to create universal NNIPs [66], given that pre-trained ANI models were able to escape the scaling relation seen in figure 1(b).

Although the scaling relation can estimate the extrapolation slope within reasonable bounds, it is unable to recover trends within the same architecture in the low-error regime. As best-performing models often have differences of force errors smaller than 5 meV $Å^{-1}$ in the 300 K test set, their extrapolation behavior may not be accurately recovered from the scaling relation due to the statistical noise in the measurements. Indeed, figure 1(c) shows the correlation between the forces RMSE computed on the 300 K split and all splits (300, 600, and 1200 K, 'high T') for several NequIP and MACE models with different hyperparameters and training methods (see tables S3 and S6), but similar testing error at the 300 K split. Although there is a positive correlation between the two RMSE metrics, the dispersion of points indicates that models with nearly the same RMSE for the 300 K data split show discrepancies in error above 10 meV $Å^{-1}$ when all splits are taken into account.

This scenario is aggravated when MD simulations are used to test the extrapolation power of the NNIPs, which may exhibit higher diversity of configurations than those used to compute the extrapolation slopes. When measuring the average simulation times for each of the models (see Methods), no clear correlation is obtained between the error on the 300 K split of 3BPA and the average (physically meaningful) simulation length (figure 1(c)). This motivates the creation of a metric that captures robustness trends in NNIPs.

3.3. Training insights derived from loss landscapes

Beyond data regularization, architectural and training choices strongly influence generalization ability of NN models. Relevant aspects include model initialization, hyperparameter optimization, choice of optimizer, batch sizes, and many others. Although the process of training NNs strongly affects their extrapolation behavior, good NN models systematically outperform their counterparts by optimizing to better local minima in the LL [67]. Thus, using these insights, we propose that *loss landscapes of NNIP models can predict their generalization ability towards unseen data despite using only the training data*. Correlations between robust generalization and loss sharpness have been observed for other NN models in the literature [41, 42], but not yet explored in the context of NNIPs.

To first verify if qualitative insights could be derived from LLs in the context of NNIPs, we investigated the behavior of the loss function around the optimized minima of the NNIP models trained to the non-noisy 3BPA dataset. To ensure the LL visualizations were statistically meaningful, we sampled 20 different orthogonal directions for each set of parameters and models, and interpolated them using the filter-normalized method described in the Methods (see also figure 2(a)). Then, we compared the LLs of the NNIP models using 2D visualizations, as often done for NN classifiers [44]. Qualitative inspection of the 2D LLs in figure 2(b) reveals the presence of weight degeneracies in the prediction of energies in models (see also figure S14 for another example in the ANI-Al dataset). This 'valley-like' landscape represents a subspace of weights leading to similar accuracy in energy [68], and reflects the interplay between energy and forces during training. These results agree with the literature regarding LLs of over-parameterized models [69], as well as the notion that physical systems often result in so-called 'sloppy' models [70, 71], and can improve trainability and interpolation [72].

Qualitative analysis of LLs also explain other factors typically found as heuristics of NNIP training. For example, the energy/forces coefficients in the final loss are often defined from hyperparameter optimization

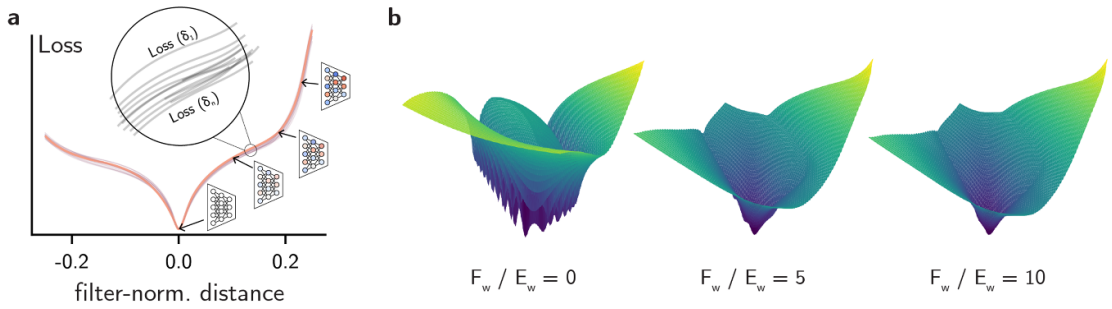

Figure 2. (a) A schematic describing the process of generating loss landscapes. Starting at the origin with a trained model, landscapes are constructed by performing a grid-sampling along a random filter-normalized direction [44] in parameter space up to a chosen maximum distance. In order to ensure that that results were consistent regardless of the choice of random sampling direction, landscapes were averaged over multiple random directions, $\delta_1 \ldots \delta_n$, with $n = 20$ usually showing only slight variations in landscape topography. (b) 2D LLs for a MACE model trained to the 3BPA dataset, generated by sampling a plane defined by two random directions. The model was trained using $E_w = 1$ (eV^{-1}) and $F_w = 1000.0$ (Å eV^{-1}), but the final landscape was re-weighted for the purpose of this figure using a fixed value of $E_w = 1.0$ and F_w increasing from 0.0 to 10.0 to demonstrate the effects of different weights on optimization.

[27, 73] and have ratios varying from 1:10 to 1:100 for energy:forces losses, but the success of higher force coefficients can be justified using LLs. In figure 2(b), we show how a higher weight on force losses leads to LLs with less saddle points around the optimized minima, thus favoring training. Although energy and forces are related and completely disentangling their effects is not achievable, the interpolation in figure 2(b) shows how mixing both can lead to better optimization landscapes for NNIPs. Based on these results, an effective training regimen would start with a relatively large weight for forces loss for a fast (and smoother) optimization of forces. Once the force errors are reasonably converged, the weight can be decreased until a desired threshold in energy errors is achieved. Similar strategies with WC and scheduling—e.g. starting with an energy:force loss weighting of 1:10, but then switch to 1000:1 in the later stages of training—can also be effective.

One limitation of visualizations of the high-dimensional LL is that different directions in weight perturbation may lead to similar losses. This results in only slight variations in landscape topography with different random directions, as noted in figure 2. This may be related to the dominance of specific layers of the model in the filter normalization technique. Figure S13 shows how the distribution of weights is non-uniform, suggesting that some layers have higher sensitivity to weight perturbation than others. As the filter normalization displaces the model weights along random directions with magnitude proportional to the norm of each filter and layer, parameters with higher weights may influence certain regions of the loss landscapes. An exception to this point is when the parameters are intertwined with functions embedded in the architecture, such as the trainable Bessel functions in NequIP. As shown in figure S6, freezing certain high-magnitude weights when generating the LLs can help flatten the landscape and remove spurious minima, emphasizing the importance of proper regularization and training regimen taking these effects into account (e.g. separate learning rates for certain layers in the model).

3.4. Loss landscapes predict extrapolation trends in NequIP

In the context of NNIPs, robust generalization has important ramifications in two distinct areas: the stability of the model in production (e.g. MD simulations), and its data efficiency during training. To probe the first of these features, we trained NequIP models using various choices of model architecture and optimization techniques (see table S1) in the low-temperature split of the 3BPA dataset. Then, we performed MD simulations in the NVT ensemble with a high temperature of 1600 K (see Methods) to ensure all models were extrapolating well beyond their training split. Furthermore, as the 3BPA benchmark already provides geometries sampled from simulations up to 1200 K, the MD trajectories at 1600 K could provide information beyond the 1200 K splits provided by the dataset. For each model in the study, 30 MD trajectories were simulated to obtain reasonable statistics on the model behavior, with simulation timelengths of up to 6 ps and a timestep of 1 fs. In principle, a model's ability to extrapolate beyond the training set can be assessed by the number of trajectories that behave in a physical way [38], as well as the average trajectory length until the model fails to extrapolate (see Methods).

Using the results from MD simulations, we investigate whether LLs can predict the extrapolation behavior and trajectory stability of models. While the test performance at low-temperature samples is unable to perform such predictions, the test error at high-temperature samples is still expected to be a reasonable predictor of such stability. Nevertheless, whereas high-temperature samples are available in the 3BPA dataset,

out-of-domain data is often not available for an arbitrary material system. In addition to being expensive to generate using ground-truth methods, sampling new configurations is rarely performed in an exhaustive way. Thus, the major advantage of extrapolation tests based on LLs is their reliance only on the training dataset. To quantify the sharpness of the LLs [42], we compute the loss entropy as described in the Methods, with T_E and T_F taken to be reasonable 'room temperature' values of the energy/force RMSEs, respectively ($T_E = 4$ meV/atom, $T_F = 40$ meV Å^{-1}). The weight α was set to 0.2 to resemble the higher force weighting used during training without ignoring the contributions of energy LLs in model stability. A full sensitivity analysis of S with respect to these parameters showed that results remained consistent around these error ranges, as long as unreasonable temperature values were not used (see figure S4).

The relationship between the model/training parameters, the MD stability, and LLs entropy are shown in figure 3(see also table S2). The analysis is grouped into four experiments (columns in figure 3) to isolate the effects of specific portions of the training procedure and model architecture, thus uncovering useful trends for NNIP practitioners. For example, considering only the distributions of time to failure from MD simulations in figure 3(b), it can be seen that rescaling the energies predicted by the model (models 'no rescaling' vs. 'rescaling') greatly improves the stability of the model, especially when the Bessel weights used by the radial basis are trainable. This behavior is reflected in the energy and forces LLs (figure 3(a)), with the LL of the model without rescaling showing considerable sharpness compared to the models using rescaled energies. Quantitative trends are also obtained when the loss entropy and test RMSE at all splits in the 3BPA dataset are computed (figure 3(c), first column, and table S3). Models with rescaling showed higher average simulation time with physical trajectories, as well as increased entropy and lower forces RMSE.

When comparing how the training regime can change the extrapolation behavior of NequIP, the results show that the AMSGrad variant of the Adam optimizer [74] leads to consistent improvements in the model extrapolation both for the two-layer and the five-layer model (figure 3(b), second and third column) compared to the baseline, which does not use AMSGrad. The improvements of simulation quality are particularly pronounced in the five-layer models, where the model trained using AMSGrad shows significant improvements in simulation stability compared to the baseline despite not using trainable Bessel functions (table S1). On the other hand, the exponential moving average (EMA) appears to degrade the extrapolation performance of the NNIPs, often leading to sharper LLs if used in isolation and with constant learning rates. These trends are reflected in the extrapolation metrics (figure 3(c)). Although the loss entropy underperforms compared to the forces RMSE in the case of the two-layer models, the trend is correctly captured for the five-layer model, where the model trained with AMSGrad showed both higher loss entropy and higher MD stability.

Finally, NequIP models can be compared according to the number of message-passing layers. Using a fixed training regime, a higher number of layers showed pronounced improvement both in the stable simulation time (figure 3(b)), as well as in the forces RMSE and loss entropy (figure 3(c)), with the trend remaining consistent across these models. Interestingly, the improvement in loss entropy for the 5-layer model is reflected mostly in the energy LL rather than the force LL (figure 3(a)). Although only forces are required when integrating the equations of motion in the NVT ensemble during an MD simulation, the wider minima in energy LL reflects the model's higher generalization capacity towards forces obtained by differentiating the energy through the NNIP architecture. This phenomenon can be explained by recognizing that the out-of-domain data often results in a horizontally shifted version of the loss landscape [47]. To explain this effect, we computed the LL of NequIP models with respect to the test sets instead of the training set (see figure S5). As expected, the test LLs show higher errors compared to the train LL, as seen in the vertical shift of the curves. However, while the forces LL only undergo a vertical shift, test energy LLs also undergo a horizontal shift, indicating that the optimal model for the training set is not necessarily optimal for predicting energies of the testing sets. Furthermore, as the forces loss is often derived from the energy in NNIPs, the propagation of these mismatches may be responsible for degradation in extrapolation performance. Thus, in general, the results in figure 3 show that architectures and optimization strategies which result in loss landscapes with higher entropy (i.e. flatter landscapes) tend to demonstrate improved stability in MD simulations that sample out-of-domain configurations.

3.5. Loss landscapes predict extrapolation trends in MACE

To confirm that the results could be extended beyond NequIP, we performed a similar study using the MACE framework. Given the differences in architecture and available code, MACE has different hyperparameters and optimizer choices than NequIP (see table S4). Nevertheless, a similar study for MACE reproduces the trends seen for NequIP, as shown in figure 4.

As seen in the case of NequIP, model stability can be greatly improved by using rescaling and AMSGrad. Also for MACE, EMA appears to lower the time-to-failure if used in isolation, with similar trends in out-of-domain RMSE, loss entropy, and MD stability seen in the NequIP case (figure 4). In addition, we also

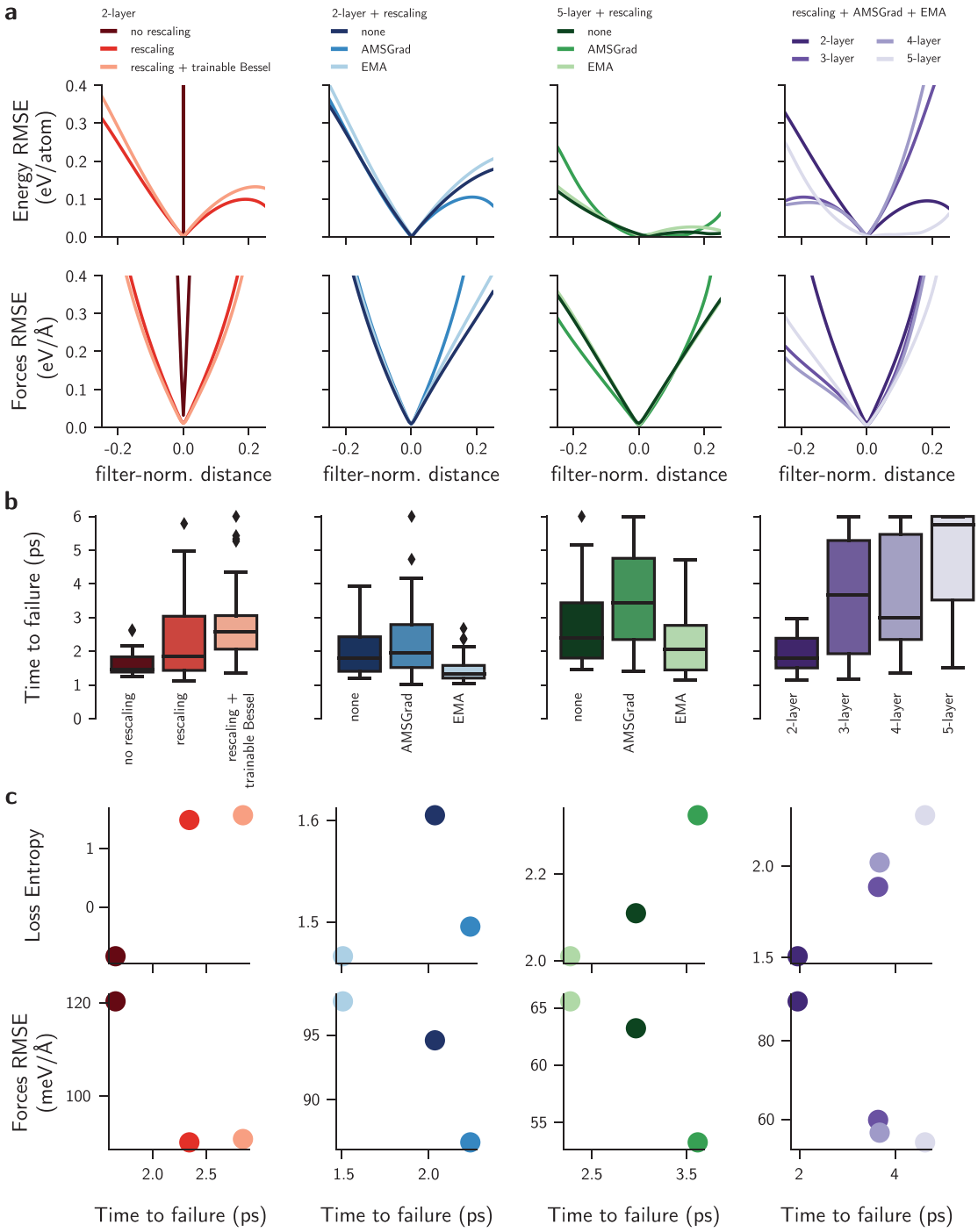

Figure 3. Analysis of LLs and MD simulation stability results for NequIP models while varying: (column 1) rescaling and Bessel weights, (column 2) optimizer settings using a model with two interaction blocks or (column 3) five interaction blocks, and (column 4) the number of interaction blocks. (a) LLs for each model (where flatter landscapes are expected to correspond to more robust models) and (b) distributions of time to failure computed over 30 MD simulations. (c) Time to failure compared to (top) the entropy of the loss landscape (as defined in the Methods) or the forces RMSE on the high-temperature test sets. The horizontal line, box, and whiskers in (b) depict the median, interquartile range, and points within 150% of the interquartile range of the distribution. Isolated diamonds are points outside of this range.

explored the effects of stochastic weight averaging (SWA) [75] and WC (column 2 of figure 4) implemented in the MACE code. Consistent with results from the use of SWA for image classification tasks [75], SWA + WC is shown to improve model generalizability, leading to higher simulation times and flatter loss landscapes. Interestingly, the model trained with SWA + WC exhibits similar force RMSE in high-temperature samples compared to the baseline, but a higher loss entropy and higher MD stability (figure 4(c)). As SWA flattens the loss landscape by design [75], implementing this strategy in other NNIP models may be a low-cost modification for improving the generalization capacity of different architectures.

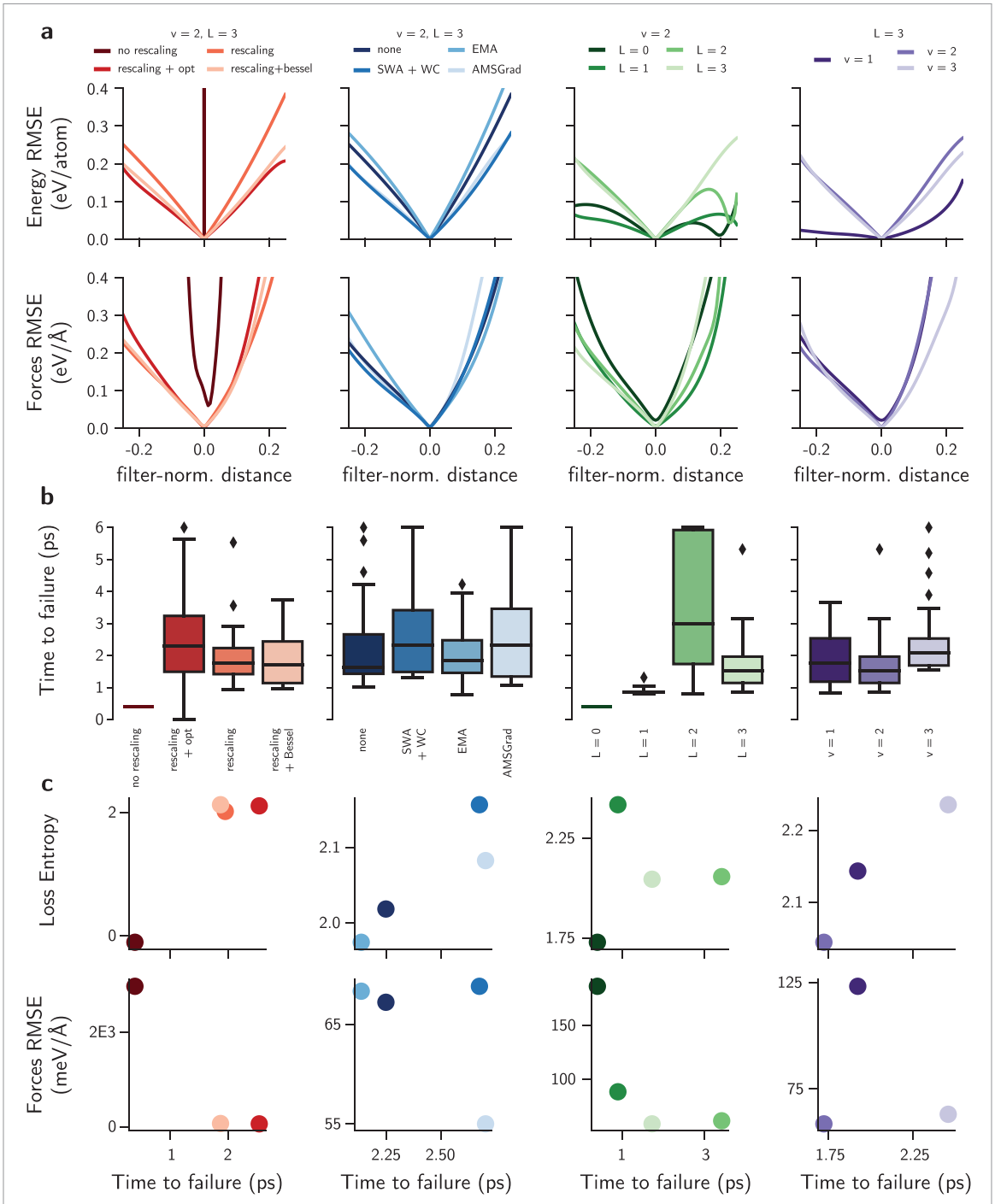

Figure 4. Analysis of loss landscapes and MD simulation stability results for MACE models while varying: (column 1) the use of rescaling; (column 2) optimizer settings using a model with a fixed body order, $\nu = 2$, and symmetry order, $L = 3$; (column 3) the symmetry order of the edge features with a fixed body order of 2; and (column 4) the body order with a fixed symmetry order of 3. (a) Loss landscapes for each model (where flatter landscapes are expected to correspond to more robust models). (b) Distributions of time-to-failure computed over 30 MD trajectories. (c) Time-to-failure compared to (top) the entropy of the loss landscape (as defined in the Methods) or the forces RMSE on all test sets of the 3BPA dataset. The horizontal line, box, and whiskers in (b) depict the median, interquartile range, and points within 150% of the interquartile range of the distribution. Isolated diamonds are points outside of this range.

The use of WC can also help the optimization process (see previous results) and lead to better energy landscapes overall.

For the MACE study, we also analyzed the relationship between the bond order of the model (column 4) and its stability in production MD simulations. In this case, the loss entropy recovers the trend in MD stability better than the forces RMSE (figure 4(c), column 4). Interestingly, despite the higher errors of the $\nu = 1, L = 3$ model compared to its $\nu = 2, L = 3$ counterpart, the former still exhibits a higher average time to failure. This observation can be traced back to the flatter energy LL (figure 4(a), column 4), as also seen in the case of NequIP.

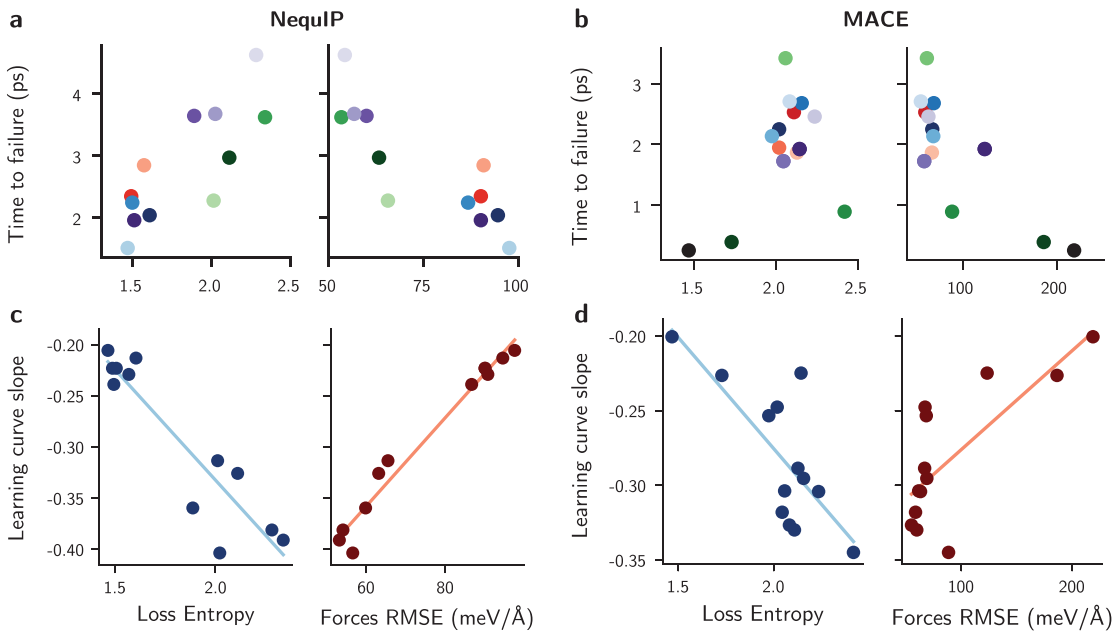

Figure 5. Loss landscape entropy and high-temperature force test RMSEs compared to (a) and (b) MD time-to-failure or (c) and (d) learning curve slope for (a) and (c) NequIP and (b) and (d) MACE models. Higher time to failure and more negative learning curve slopes correspond to more robust and data-efficient models, respectively. Colors for points in (a) and (b) were chosen to match the colors in figures 3 and 4 respectively. Lines in (c) and (d) represent linear fits to the data.

Finally, the models were compared according to the symmetry order of the edge features (column 3). In general, increasing the value of L from $L = 0$ to $L = 2$ led to more stable simulations, although the model with $L = 3$ shows a degradation of the production behavior. Nevertheless, the LLs do not follow the expected trends explained so far in this work. We believe the filter normalization technique discussed in the Methods cannot provide comparable LLs upon changes of L in MACE. As this parameter affects the tensor order of some model components [29], the number of parameters in these particular filters grows with $\mathcal{O}(N^L)$, which may be affecting the filter normalization. In contrast, adding message-passing layers with similar numbers of parameters, as in NequIP, allows the filter normalization technique to create comparable loss landscapes. This result shows some limitations of the loss entropy when comparing models with different architectures or hyperparameters, and may require different strategies for computing them in the future.

3.6. Loss landscapes and data efficiency

Since extrapolation power and data efficiency are conceptually related, a natural extension of these results is to verify whether the loss landscape entropy can also be used to predict the data efficiency of a model during training. As with many other applications of deep learning, generating training data is often one of the most costly steps in the development process. Especially considering the 'denoising' effects of NNIPs demonstrated in this work, identifying training techniques and model architectures that lead to more data-efficient training has the potential to greatly reduce the computational cost of building new NNIPs.

To test whether loss entropy can predict the data efficiency of models, we computed the learning curves for NequIP and MACE models with the hyperparameters selected in the previous sections. This is achieved by training models with the specified training parameters and architectures to the same subsets of the 3BPA training set using only 25, 125, 250, or 500 samples from the 300 K split (tables S3 and S6). Following previous work [27], the slopes of the learning curves were then computed by fitting the line $\log n = m \log \varepsilon + b$ to the number of training samples n and the force RMSEs ε calculated using all test splits (300, 600, and 1200 K), and comparing the slopes m. Consistent with the results from the previous sections (combined in figures 5(a) and (b)), the high-temperature force RMSEs (i.e. the points from the learning curves using $n = 500$) are good predictors of the learning curve slope. In addition, figures 5(c) and (d) show that the loss entropy also has a high correlation with the slope of the learning curve despite being computed using only the training set. As discussed previously, these correlations are less explicit for the case of MACE models, and thus show that none of the metrics is truly universal for predicting stability in simulations and learning curve slopes. Nevertheless, these results further demonstrate that loss landscapes provide information on the extrapolation behavior of NNIPs despite being derived only from the training set.

4. Conclusion

In this work, we motivate the need for additional metrics beyond RMSE by showing that in-domain errors fail to predict model stability in the high-accuracy regime despite large-scale trends in extrapolation behavior and NNIP robustness to noise. We propose the use of loss entropy as a metric for quantifying the extrapolation behavior, and demonstrate that it correlates well with out-of-distribution error and stability in production simulations. Using large studies with NequIP and MACE, we show how models containing flatter loss landscapes exhibit better extrapolation behavior, and how different training parameters can be used to achieve them. For example, rescaling, AMSGrad, and SWA were shown to increase the loss entropy and MD stability, and may be important tools when training NNIP models. Similarly, models with similar test error can be distinguished by their energy loss landscape, with models displaying broader minima performing better in extrapolation tasks. Future studies can address shortcomings of loss landscape visualizations in NNIP by analyzing how filter-normalization can be made more suitable for NNIP architectures, or to better isolate the effects of key architectural changes in the loss. Nevertheless, these results can better inform the development of new model architecture and optimization strategies in NNIPs, facilitating their use in general materials simulation.

Data availability statements

The datasets used to train the models in this work were obtained directly from their original sources: https://github.com/davkovacs/BOTNet-datasets/tree/main/dataset_3BPA (3BPA) and https://github.com/atomistic-ml/ani-al (ANI-Al). The package `ip_explorer` is available at https://github.com/jvita/ip_explorer. Loss landscape calculations were performed using the the code from the public package https://github.com/marcellodebernardi/loss-landscapes. Training codes for NequIP and MACE are available from their original authors as described in appendix A in the supporting information.

The data that supports the findings of this study are openly available at the following URL/DOI: https://github.com/jvita/data_efficiency_in_IAPS [76].

Acknowledgments

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344, funded by the Laboratory Directed Research and Development (LDRD) Program at LLNL under project tracking code 22-ERD-055. Computing resources were provided by Livermore Computing. The authors thank Vincenzo Lordi, the Quantum Simulations Group at LLNL, and Simon Batzner for the discussions. We also thank Ilyes Batatia for the support with the MACE code. Manuscript released as `LLNL-JRNL-845001`.

Author contributions

Conceptualization: DSK
Methodology: DSK, JAV
Software: DSK, JAV
Validation: DSK, JAV
Investigation: DSK, JAV
Data Curation: DSK, JAV
Writing—Original Draft: DSK, JAV
Writing—Review & Editing: DSK, JAV
Visualization: DSK, JAV
Supervision: DSK

Conflict of interest

The authors declare that they have no competing interests.

ORCID iDs

Joshua A Vita https://orcid.org/0000-0001-9191-055X
Daniel Schwalbe-Koda https://orcid.org/0000-0001-9176-0854

References

[1] Bartók A P, De S, Poelking C, Bernstein N, Kermode J R, Csányi G and Ceriotti M 2017 Machine learning unifies the modeling of materials and molecules *Sci. Adv.* **3** e1701816
[2] Butler K T, Davies D W, Cartwright H, Isayev O and Walsh A 2018 Machine learning for molecular and materials science *Nature* **559** 547–55
[3] Schmidt J, Marques M R G, Botti S and Marques M A L 2019 Recent advances and applications of machine learning in solid-state materials science *npj Comput. Mater.* **5** 83
[4] Keith J A, Vassilev-Galindo V, Cheng B, Chmiela S, Gastegger M, Müller K-R and Tkatchenko A 2021 Combining machine learning and computational chemistry for predictive insights into chemical systems *Chem. Rev.* **121** 9816–72
[5] Behler J 2015 Constructing high-dimensional neural network potentials: a tutorial review *Int. J. Quantum Chem.* **115** 1032
[6] Mueller T, Hernandez A and Wang C 2020 Machine learning for interatomic potential models *J. Chem. Phys.* **152** 050902
[7] Manzhos S and Carrington T Jr 2021 Neural network potential energy surfaces for small molecules and reactions *Chem. Rev.* **121** 10187–217
[8] Chmiela S, Tkatchenko A, Sauceda H E, Poltavsky I, Schütt K T and Müller K-R 2017 Machine learning of accurate energy-conserving molecular force fields *Sci. Adv.* **3** e1603015
[9] Unke O T, Chmiela S, Sauceda H E, Gastegger M, Poltavsky I, Schütt K T, Tkatchenko A and Müller K-R 2021 Machine learning force fields *Chem. Rev.* **121** 10142–86
[10] Bartók A P, Payne M C, Kondor R and Csányi G 2010 Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons *Phys. Rev. Lett.* **104** 136403
[11] Christensen A S, Bratholm L A, Faber F A and Anatole von Lilienfeld O 2020 FCHL revisited: Faster and more accurate quantum machine learning *J. Chem. Phys.* **152** 044107
[12] Deringer V L, Bartók A P, Bernstein N, Wilkins D M, Ceriotti M and Csányi G 2021 Gaussian process regression for materials and molecules *Chem. Rev.* **121** 10073–141
[13] Behler J and Parrinello M 2007 Generalized neural-network representation of high-dimensional potential-energy surfaces *Phys. Rev. Lett.* **98** 146401
[14] Behler J, Martoňák R, Donadio D and Parrinello M 2008 Metadynamics simulations of the high-pressure phases of silicon employing a high-dimensional neural network potential *Phys. Rev. Lett.* **100** 185501
[15] Cheng B, Engel E A, Behler J, Dellago C and Ceriotti M 2019 *Ab initio* thermodynamics of liquid and solid water *Proc. Natl Acad. Sci.* **116** 1110–5
[16] Westermayr J, Gastegger M, Vörös D, Panzenboeck L, Joerg F, González L and Marquetand P 2022 Deep learning study of tyrosine reveals that roaming can lead to photodamage *Nat. Chem.* **14** 914–9
[17] Behler J 2011 Atom-centered symmetry functions for constructing high-dimensional neural network potentials *J. Chem. Phys.* **134** 074106
[18] Smith J S, Isayev O and Roitberg A E 2017 ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost *Chem. Sci.* **8** 3192–203
[19] Huan T D, Batra R, Chapman J, Krishnan S, Chen L and Ramprasad R 2017 A universal strategy for the creation of machine learning-based atomistic force fields *npj Comput. Mater.* **3** 1–8
[20] Zhang L, Han J, Wang H, Car R and Weinan E 2018 Deep potential molecular dynamics: a scalable model with the accuracy of quantum mechanics *Phys. Rev. Lett.* **120** 143001
[21] Wood M A and Thompson A P 2018 Extending the accuracy of the SNAP interatomic potential form *J. Chem. Phys.* **148** 241721
[22] Drautz R 2019 Atomic cluster expansion for accurate and transferable interatomic potentials *Phys. Rev.* B **99** 014104
[23] Gilmer J, Schoenholz S S, Riley P F, Vinyals O and Dahl G E 2017 Neural message passing for quantum chemistry (arXiv:1704.01212)
[24] Kondor R, Lin Z and Trivedi S 2018 Clebsch-Gordan nets: a fully fourier space spherical convolutional neural network (arXiv:1806.09231)
[25] Thomas N *et al* 2018 Tensor field networks: rotation- and translation-equivariant neural networks for 3D point clouds (arXiv:1802.08219)
[26] Anderson B, Hy T-S and Kondor R 2019 Cormorant: covariant molecular neural networks (arXiv:1906.04015)
[27] Batzner S, Musaelian A, Sun L, Geiger M, Mailoa J P, Kornbluth M, Molinari N, Smidt T E and Kozinsky B 2022 E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials *Nat. Commun.* **13** 2453
[28] Batatia I *et al* 2022 The design space of E(3)-equivariant atom-centered interatomic potentials (arXiv:2205.06643)
[29] Batatia I, Kovács D P, Simm G N C, Ortner C and Csányi G 2022 MACE: higher order equivariant message passing neural networks for fast and accurate force fields (arXiv:2206.07697)
[30] Musaelian A *et al* 2022 Learning local equivariant representations for large-scale atomistic dynamics (arXiv:2204.05249)
[31] Zuo Y *et al* 2020 Performance and cost assessment of machine learning interatomic potentials *J. Phys. Chem.* A **124** 731–45
[32] Kovács D P, Oord C van der, Kucera J, Allen A E A, Cole D J, Ortner C and Csányi G 2021 Linear atomic cluster expansion force fields for organic molecules: beyond RMSE *J. Chem. Theory Comput.* **17** 7696–711
[33] Fu X *et al* 2022 Forces are not enough: benchmark and critical evaluation for machine learning force fields with molecular simulations (arXiv:2210.07237)
[34] Stocker S, Gasteiger J, Becker F, Günnemann S and Margraf J T 2022 How robust are modern graph neural network potentials in long and hot molecular dynamics simulations? *Mach. Learn.: Sci. Technol.* **3** 045010
[35] Morrow J D, Gardner J L and Deringer V L 2022 How to validate machine-learned interatomic potentials (arXiv:2211.12484)
[36] Vita J A and Trinkle D R 2021 Exploring the necessary complexity of interatomic potentials *Comput. Mater. Sci.* **200** 110752
[37] Wellawatte G P, Hocky G M and White A D 2023 Neural potentials of proteins extrapolate beyond training data *ChemRxiv Preprint* (https://doi.org/10.26434/chemrxiv-2022-41f04-v3)
[38] Schwalbe-Koda D, Tan A R and Gómez-Bombarelli R 2021 Differentiable sampling of molecular geometries with uncertainty-based adversarial attacks *Nat. Commun.* **12** 5104
[39] Hochreiter S and Schmidhuber J 1994 Simplifying neural nets by discovering flat minima *Advances in Neural Information Processing Systems* vol 7
[40] Hinton G E and Van Camp D 1993 Keeping the neural networks simple by minimizing the description length of the weights *Proc. 6th Annual Conf. on Computational Learning Theory* pp 5–13
[41] Chaudhari P *et al* 2016 Entropy-SGD: biasing gradient descent into wide valleys (arXiv:1611.01838)

[42] Jiang Y, Neyshabur B, Mobahi H, Krishnan D and Bengio S 2019 Fantastic generalization measures and where to find them (arXiv:1912.02178)
[43] Goodfellow I J, Vinyals O and Saxe A M 2014 Qualitatively characterizing neural network optimization problems (arXiv:1412.6544)
[44] Li H, Xu Z, Taylor G, Studer C and Goldstein T 2017 Visualizing the loss landscape of neural nets (arXiv:1712.09913)
[45] Ballard A J, Das R, Martiniani S, Mehta D, Sagun L, Stevenson J D and Wales D J 2017 Energy landscapes for machine learning *Phys. Chem. Chem. Phys.* **19** 12585–603
[46] Verpoort P C, Lee A A and Wales D J 2020 Archetypal landscapes for deep neural networks *Proc. Natl Acad. Sci.* **117** 21857–64
[47] Keskar N S, Mudigere D, Nocedal J, Smelyanskiy M and Tang P T P 2016 On large-batch training for deep learning: generalization gap and sharp minima (arXiv:1609.04836)
[48] Neyshabur B, Bhojanapalli S, McAllester D and Srebro N 2017 Exploring generalization in deep learning (arXiv:1706.08947)
[49] Im D J, Tao M and Branson K 2016 An empirical analysis of the optimization of deep network loss surfaces (arXiv:1612.04010)
[50] Nguyen Q and Hein M 2017 The loss surface of deep and wide neural networks (arXiv:1704.08045)
[51] Smith L N and Topin N 2017 Exploring loss function topology with cyclical learning rates (arXiv:1702.04283)
[52] Baldassi C, Pittorino F and Zecchina R 2020 Shaping the learning landscape in neural networks around wide flat minima *Proc. Natl Acad. Sci.* **117** 161–70
[53] Gasteiger J, Becker F and Günnemann S 2021 GemNet: universal directional graph neural networks for molecules (arXiv:2106.08903)
[54] Gasteiger J, Groß J and Günnemann S 2020 Directional message passing for molecular graphs (arXiv:2003.03123)
[55] Lubbers N, Smith J S and Barros K 2018 Hierarchical modeling of molecular energies using a deep neural network *J. Chem. Phys.* **148** 241715
[56] Haghighatlari M *et al* 2021 NewtonNet: a Newtonian message passing network for deep learning of interatomic potentials and forces (arXiv:2108.02913)
[57] Schütt K T, Sauceda H E, Kindermans P-J, Tkatchenko A and Müller K-R 2018 SchNet—a deep learning architecture for molecules and materials *J. Chem. Phys.* **148** 241722
[58] Schütt K T, Unke O T and Gastegger M 2021 Equivariant message passing for the prediction of tensorial properties and molecular spectra (arXiv:2102.03150)
[59] Hu W *et al* 2021 ForceNet: a graph neural network for large-scale quantum calculations (arXiv:2103.01436)
[60] Smith J S *et al* 2021 Automated discovery of a robust interatomic potential for aluminum *Nat. Commun.* **12** 1–13
[61] Berendsen H J, Postma J P M, Van Gunsteren W F, DiNola A and Haak J R 1984 Molecular dynamics with coupling to an external bath *J. Chem. Phys.* **81** 3684–90
[62] Larsen A H *et al* 2017 The atomic simulation environment—a Python library for working with atoms *J. Phys.: Condens. Matter* **29** 273002
[63] Zhang C, Bengio S, Hardt M, Recht B and Vinyals O 2016 Understanding deep learning requires rethinking generalization (arXiv:1611.03530)
[64] Devereux C, Smith J S, Huddleston K K, Barros K, Zubatyuk R, Isayev O and Roitberg A E 2020 Extending the applicability of the ANI deep learning molecular potential to sulfur and halogens *J. Chem. Theory Comput.* **16** 4192–202
[65] Abnar S, Dehghani M, Neyshabur B and Sedghi H 2021 Exploring the limits of large scale pre-training (arXiv:2110:02095)
[66] Chen C and Ong S P 2022 A universal graph deep learning interatomic potential for the periodic table *Nat. Comput. Sci.* **2** 718–28
[67] Santurkar S, Tsipras D, Ilyas A and Madry A 2018 How does batch normalization help optimization? *Advances in Neural Information Processing Systems* vol 31, ed S Bengio *et al* (Curran Associates, Inc.)
[68] Draxler F, Veschgini K, Salmhofer M and Hamprecht F 2018 Essentially no barriers in neural network energy landscape *Int. Conf. on Machine Learning* (PMLR) pp 1309–18
[69] Liu C, Zhu L and Belkin M 2022 Loss landscapes and optimization in over-parameterized non-linear systems and neural networks *Appl. Comput. Harmon. Anal.* **59** 85–116
[70] Gutenkunst R N, Waterfall J J, Casey F P, Brown K S, Myers C R and Sethna J P 2007 Universally sloppy parameter sensitivities in systems biology models *PLoS Comput. Biol.* **3** e189
[71] Kurniawan Y, Petrie C L, Williams K J, Transtrum M K, Tadmor E B, Elliott R S, Karls D S and Wen M 2022 Bayesian, frequentist and information geometric approaches to parametric uncertainty quantification of classical empirical interatomic potentials *J. Chem. Phys.* **156** 214103
[72] Bubeck S and Sellke M 2021 A universal law of robustness via isoperimetry (arXiv:2105.12806)
[73] Schütt K T, Arbabzadah F, Chmiela S, Müller K R and Tkatchenko A 2017 Quantum-chemical insights from deep tensor neural networks *Nat. Commun.* **8** 13890
[74] Reddi S J, Kale S and Kumar S 2018 On the convergence of adam and beyond *Int. Conf. on Learning Representations* (available at: https://openreview.net/forum?id = ryQu7f-RZ)
[75] Izmailov P, Podoprikhin D, Garipov T, Vetrov D and Wilson A G 2018 Averaging weights leads to wider optima and better generalization (arXiv:1803.05407)
[76] Vita Joshua and Schwalbe-Koda Daniel 2023 Data for: "Data efficiency and extrapolation trends in neural network interatomic potentials" (available at: https://github.com/jvita/data_efficiency_in_IAPS)