Stability-Aware Training of Machine Learning Force Fields with Differentiable Boltzmann Estimators

Anonymous authors
Paper under double-blind review

Abstract

Machine learning force fields (MLFFs) are an attractive alternative to *ab-initio* methods for molecular dynamics (MD) simulations. However, they can produce unstable simulations, limiting their ability to model phenomena occurring over longer timescales and compromising the quality of estimated observables. To address these challenges, we present Stability-Aware Boltzmann Estimator (StABlE) Training, a multi-modal training procedure which leverages joint supervision from reference quantum-mechanical calculations and system observables. StABlE Training iteratively runs many MD simulations in parallel to seek out unstable regions, and corrects the instabilities via supervision with a reference observable. We achieve efficient end-to-end automatic differentiation through MD simulations using our Boltzmann Estimator, a generalization of implicit differentiation techniques to a broader class of stochastic algorithms. Unlike existing techniques based on active learning, our approach requires no additional *ab-initio* energy and forces calculations to correct instabilities. We demonstrate our methodology across organic molecules, tetrapeptides, and condensed phase systems, using three modern MLFF architectures. StABlE-trained models achieve significant improvements in simulation stability, data efficiency, and agreement with reference observables. By incorporating observables into the training process alongside first-principles calculations, StABlE Training can be viewed as a general semi-empirical framework applicable across MLFF architectures and systems. This makes it a powerful tool for training stable and accurate MLFFs, particularly in the absence of large reference datasets.

1 Introduction

Molecular dynamics (MD) simulation is a staple method of computational science, enabling high-resolution spatiotemporal modeling of atomistic systems throughout biology, chemistry, and materials science (Frenkel & Smit, 2001).While the atomic forces needed for MD simulation can be obtained on-the-fly via *ab-initio* quantum-mechanical (QM) calculations (Car & Parrinello, 1985), this is prohibitively expensive for realistic length and time scales (Friesner, 2005). Machine learning force fields (MLFFs) have recently emerged as a promising option to serve as surrogate models for QM calculations, demonstrating the ability to capture complex many-body interactions, and in some cases transfer flexibly across chemical space (Schütt et al., 2018; Hu et al., 2021; Liu et al., 2022; Gasteiger et al., 2021; 2020; Batzner et al., 2022; Musaelian et al., 2022; Batatia et al., 2022; Schütt et al., 2021). Graph neural network (GNN)-based MLFFs trained on large *ab-initio* datasets are increasingly being used to model challenging and important chemical systems with favorable results (Merchant et al., 2023; Kozinsky et al., 2023; Chen & Ong, 2022; Schaarschmidt et al., 2022; Takamoto et al., 2022; Majewski et al., 2023; Charron et al., 2023; Batatia et al., 2023; Kovács et al., 2023; Shoghi et al., 2023; Gasteiger et al., 2022; Deng et al., 2023).

MD simulations aim to accurately estimate system observables like the radial distribution function, virial stress tensor, and diffusivity coefficient. This often requires long simulation to fully explore the underlying PES and minimize sampling error. Unfortunately, MLFFs are known to produce unstable simulations, meaning that they can irreversibly enter unphysical regions of phase space (e.g., a bond breaking event in a non-reactive system at low temperature) (Fu et al., 2022; Stocker et al., 2022; Vita & Schwalbe-Koda, 2023; Bihani et al., 2023; Morrow et al., 2023; Wang et al., 2023b). Sampling of such regions often quickly leads to unrecoverable simulation collapse can lead to inaccuracies in computed observables as the MLFF gradually drifts far from the distribution of its training data. In extreme cases, instabilities can lead to unrecoverable simulation collapse, in which case As a result, computed observables may have high statistical error due to insufficient sampling, or be altogether inaccurate. This In either case, instability

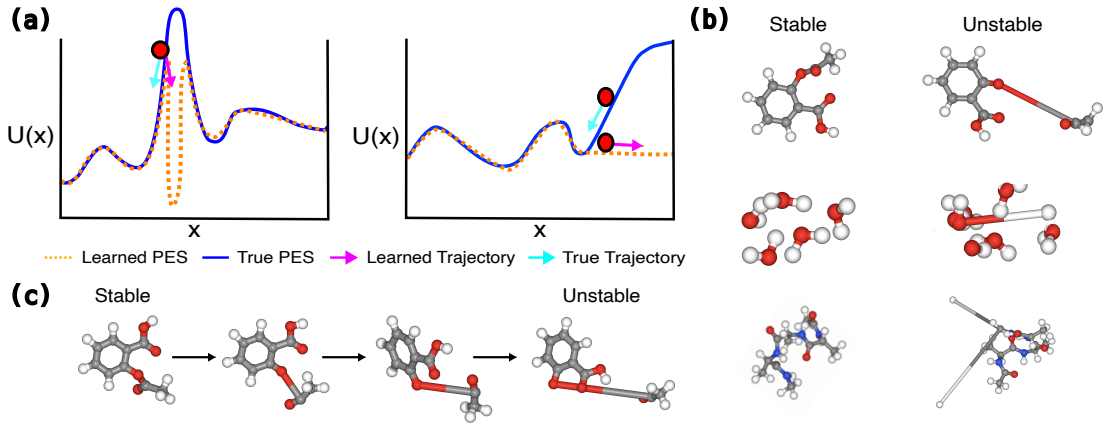

Figure 1: Machine learning force field (MLFF) failure modes. (a) Illustrative examples of true and learned potential energy surfaces (PES) and resulting dynamics for unstable MLFFs. MLFFs can be accurate in approximating most much of the PES, but contain localized "holes" regions where energy and forces estimates deviate significantly from the true PES, leading to sampling of highly unphysical regimes. As a result, observables computed from MD simulation may be inaccurate biased by the oversampling of unphysical states, or have high statistical error in the extreme case of unrecoverable simulation collapse. **(b)** Examples of stable versus unstable configurations sampled by MLFFs during molecular dynamics simulation of systems considered in this work. **(c)** Selected states from an unstable MD trajectory of aspirin.

can limit the ability of MLFF-based MD simulations to investigate long-timescale phenomena like ion diffusion and protein folding, as well as rare events that may require extensive sampling to encounter. Figure 1 illustrates typical MLFF instability behaviors and provides examples of unstable structures sampled during MD simulation.

Insufficient coverage of phase space and inaccuracies within the reference datasets utilized for training MLFFs have been suggested as factors contributing to instability Simulation instability has been shown to have an unreliable correlation with the energy and force error metrics typically used to train and evaluate MLFFs (Fu et al., 2022; Stocker et al., 2022; Bihani et al., 2023). Recent works have introduced alternative simulation-based objectives, such as localized (Wang et al., 2023b) or reweighted (Ge et al., 2024) energy and force errors, to achieve better downstream stability. However, these approaches do not allow the MLFF to visit new configurations, limiting the distribution over which it is trained and thus the potential improvement. Alternatively, expanding the phase space coverage of the dataset can be an effective way to combat MLFF instability (Fu et al., 2022; Stocker et al., 2022; Vita & Schwalbe-Koda, 2023; Bihani et al., 2023; Morrow et al., 2023; Wang et al., 2023b). Expanding the dataset requires additional QM calculations, and improving its accuracy requires more exact QM approaches (Kümmel, 2002). Both options are very computationally expensive, particularly for larger systems like macromolecules. This is typically accomplished via active learning (Smith et al., 2018; Vandermause et al., 2020; Schran et al., 2020; Lin et al., 2021; Kulichenko et al., 2023) approaches, where new atomistic configurations are selected, *ab-initio* QM calculations are performed, and the MLFF is retrained on the expanded dataset. However, these techniques rely on performing additional *ab-initio* calculations to expand the dataset. With MLFFs being trained on increasingly diverse datasets (Shoghi et al., 2023; Batatia et al., 2023; Kovács et al., 2023) and larger atomistic systems, the expense of these calculations may hamper the practicality of active learning workflows. This suggests the need for additional sources of information beyond energies and forces to train stable MLFFs.

In this work, we bridge this gap by using both system observables and *ab-initio* QM data to improve MLFF stability. We introduce **Stability-Aware Boltzmann Estimator (StABlE) Training**, a procedure designed to produce MLFFs that are both stable and accurate. The core idea behind StABlE is to which uses efficient, parallelized MD simulations to rapidly explore regions of molecular phase space where the MLFF becomes unstable, followed by a targeted refinement of these regions using reference system observables. The key to efficient and numerically stable training lies in the Boltzmann Estimator, which enables end-to-end gradient-based learning without backpropagating directly through MD simulations. We also introduce a localized version of the Boltzmann Estimator, which enables targeted refinement of local instabilities. This is particularly important for stabilizing simulations of large, condensed-phase

systems. StABlE Training is a self-contained, efficient process that leverages learning signals from both reference observables and existing QM data, with no reliance on performing additional QM reference calculations.

We demonstrate StABlE Training on three systems and MLFF architectures: 1) simulation of aspirin with SchNet (Schütt et al., 2018), 2) simulation of the Ac-Ala3-NHMe tetrapeptide with NequIP (Batzner et al., 2022), and 3) simulation of an all-atom water system with GemNet-T (Gasteiger et al., 2021). Relative to MLFFs trained solely on energies and forces, our StABlE-trained models produce significantly more stable MD simulations, recover observables more accurately, exhibit better generalization to unseen simulation temperatures, and outperform models trained on 50 times larger, labeled datasets. Our results suggest that utilizing both quantum-mechanical and observable-based modalities is required to fully exploit the available learning signal in reference datasets and train stable and accurate MLFFs. To our knowledge, StABlE Training is the first method that combines these data modalities to improve the stability of neural network potentials in MD simulations.

2 Preliminaries

Molecular Dynamics. Molecular dynamics simulation is used to evolve the positions and momenta of an atomistic system. Given a system of N atoms, its state at time t is defined by $\Gamma(t) = \{r(t), p(t)\}$, where $r(t), p(t) \in \mathbb{R}^{N\times 3}$ are the position and momenta of the atoms. We assume the system has a scalar-valued Hamiltonian function $\mathcal{H} : \mathbb{R}^{N\times 3} \times \mathbb{R}^{N\times 3} \to \mathbb{R}$ of the form $\mathcal{H}(\Gamma) = \sum_{i=1}^{N} \frac{p^{(i)^2}}{2m^{(i)}} + U(r)$ where $U : \mathbb{R}^{N\times 3} \to \mathbb{R}$ is a potential energy function and $m^{(i)}$ and $p^{(i)}$ are the mass and momentum of atom i. By updating in the direction of the per-atom forces $-\frac{\partial U}{\partial r}$ using a numerical integration scheme such as Langevin dynamics (Bussi & Parrinello, 2007), a sequence of K simulation states $\{\Gamma(t)\}_{t=0}^{K}$ is produced.

Machine Learning Force Fields. A MLFF is a function approximator which learns a potential energy U_θ and forces $F_\theta = -\nabla_r U_\theta = (F_\theta^{(1)}, ..., F_\theta^{(n)})$ from QM reference data. MLFFs are trained to minimize the following regression loss, with supervision from a dataset of reference energies and forces $\mathcal{D}_{train} = \{(r_i, U_i, F_i)\}_{i=1}^{N}$.

$$\mathcal{L}_{QM} = \frac{1}{N}\sum_{i=1}^{N}\left[\lambda_U |U_i(\Gamma) - U_\theta(\Gamma)|^2 + \lambda_F \sum_{j=1}^{n} \|F_i^{(j)}(\Gamma) + \nabla_{r^{(j)}} U_\theta(\Gamma)\|_2^2\right] \tag{1}$$

System Observables. Observables, $g(\Gamma(t))$, characterize the state of a MD simulation at time t, and relate to macroscopic properties or experimental measurements of the system. Examples include the radial and angular distribution functions, velocity autocorrelation function, and diffusivity coefficient. Observables can be computed conveniently as an empirical average over states from a MD simulation. This is justified by the ergodicity hypothesis, under which a time-average over a sufficiently long simulation converges to a distributional average over the Boltzmann distribution. More details on the observables used in this work can be found in Supplementary Section A.4.

Training MLFFs with Observables. Observables have been used extensively in the historical development of potentials (Cornell et al., 1995; Marrink et al., 2007; Li et al., 2018), and more recently for MLFFs (Thaler & Zavadlav, 2021; Wang et al., 2020) in the context of condensed-phase (Wang et al., 2023a) and titanium (Röcken & Zavadlav, 2023) systems, enhanced sampling of rare events (Šípka et al., 2023), and protein folding simulation (Ingraham et al., 2019; Navarro et al., 2023; Kolloff & Olsson, 2023). Training with observables requires an efficient way to compute gradients through MD simulations while avoiding numerical instability and memory limitations. The approach is appealing due to the lack of reliance on expensive *ab-initio* quantum mechanical calculations, and the possibility of improved empirical consistency in settings where the underlying *ab-initio* method may be unreliable (Cheetham & Seshadri, 2024). However, past works do not consider MLFFs with stability problems, and dispense entirely with using QM data, thus losing the first-principles guarantees of matching an *ab-initio* PES.

3 Methods

We present StABlE Training, our proposed procedure to train stable and accurate MLFFs by leveraging both reference quantum mechanical data and system observables. Our approach is enabled by Boltzmann Estimators, which allows gradient-based optimization of MLFFs based on system observables.

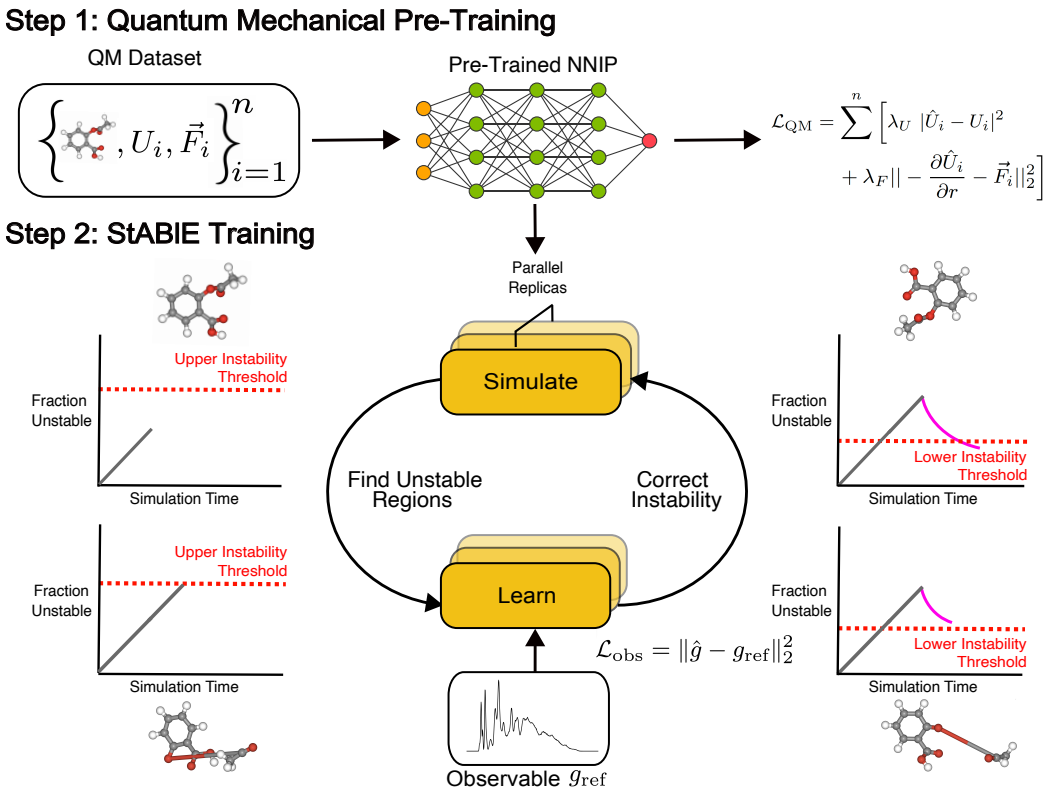

Figure 2: Schematic of Stability-Aware Boltzmann Estimator (StABlE) Training procedure. Our proposed StABlE Training procedure begins with conventional pre-training on a small reference dataset of QM calculations. This dataset remains fixed throughout the procedure, and is never expanded with new calculations. Upon convergence of pre-training, StABlE alternates between two main phases, simulation and learning. In the simulation phase, we perform many molecular dynamics simulations in parallel with the MLFF and find regions of instability. When a sufficient fraction of simulations become unstable, we enter the learning phase, where the MLFF is further trained to match known system observables from *ab-initio* simulation or experiment. Gradients are computed efficiently through the MD simulation via our Boltzmann Estimator. After a sufficient reduction in the portion of unstable trajectories, we re-enter the simulation phase, and repeat the training cycle until a predetermined computational budget is reached.

3.1 Boltzmann Estimator

Reference system observables can be estimated from high-fidelity MD simulations or experimental measurements. Since observables are linked to the MLFF parameters through a molecular simulation, training with this source of information requires a reliable way to optimize through MD trajectories.

Formally, we define $g(\Gamma)$ to be a vector-valued observable of a state Γ, and g_{ref} to be the reference value of the observable. To train our MLFF to match a reference observable, we minimize the following loss function:

$$\mathcal{L}_{\mathrm{obs}}(\theta) \stackrel{\mathrm{def}}{=} \|\mathbb{E}_{\Gamma \sim P_\theta(\Gamma)}[g(\Gamma)] - g_{\mathrm{ref}}\|_2^2, \tag{2}$$

where $P_\theta(\Gamma)$ is the equilibrium distribution induced by the MLFF U_θ. In this work, we primarily consider systems with a fixed volume, temperature, and number of particles, corresponding to the canonical (NVT) ensemble. We note, however, that the estimator is readily applicable to other ensembles, including the isothermal/isobaric (NPT) and grand canonical (μVT) ensembles (see Supplementary Sections A.2 and A.10 for details). In the canonical ensemble, microstates obey a Boltzmann distribution, $P_\theta(\Gamma) = \exp\left(-\frac{1}{k_B T}\mathcal{H}_\theta(\Gamma)\right)/C(\theta)$, where $\mathcal{H}_\theta(\Gamma)$ is the Hamiltonian, T is the sampling temperature, k_B is Boltzmann's constant, and $C(\theta) = \int \exp(-\frac{1}{k_B T}\mathcal{H}_\theta(\Gamma'))\mathrm{d}\Gamma'$

is the normalizing partition function. MD simulation is required at each training iteration to sample from this distribution and estimate the loss.

Optimizing the MLFF requires computing gradients of Equation 2 with respect to θ. The required gradient, $\nabla_\theta \mathcal{L}_{\text{obs}}$, can be decomposed via the chain rule as follows:

$$\nabla_\theta \mathcal{L}_{\text{obs}}^\top = \frac{\partial \mathcal{L}_{\text{obs}}}{\partial \mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)]} \frac{\partial \mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)]}{\partial \theta} \quad = 2(\mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)] - g_{\text{ref}})^\top \frac{\partial \mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)]}{\partial \theta}. \tag{3}$$

The non-trivial quantity to compute is the Jacobian, $\partial \mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)]/\partial\theta$. One way to estimate it is by using a chain rule expansion that corresponds to each step of the unrolled MD simulation (Wang et al., 2020). The adjoint method (Chen et al., 2019) can be used to limit the memory footprint at the expense of increased computation, but can still lead to numerical instability for long trajectories (Wang et al., 2023a; Šípka et al., 2023) (see Supplementary Section A.13 for an empirical comparison with direct and adjoint-based backpropagation).

We can avoid direct backpropagation through the simulation by noting that the equilibrium state distribution $P_\theta(\Gamma)$ is independent of the algorithm (i.e., MD integrator) used to sample the distribution. This is analogous to implicit differentiation techniques for differentiable optimization (Amos & Kolter, 2017; Gould et al., 2016; Ren et al., 2022; Blondel et al., 2022; Négiar et al., 2023), in which the solution to an optimization problem is decoupled from the numerical solver used to obtain it. We leverage the known Boltzmann form of $P_\theta(\Gamma)$ to construct an unbiased estimator of the Jacobian.

Definition 1 (N-sample Boltzmann estimator)**.** *Given N independent samples $\Gamma_1,\ldots,\Gamma_N$ from a Boltzmann distribution $P_\theta(\Gamma)$, we define the N-sample Boltzmann estimator $\mathcal{E}(\Gamma_1,\ldots,\Gamma_N)$ of the Jacobian $\partial \mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)]/\partial\theta$ as,*

$$\mathcal{E}(\Gamma_1,\ldots,\Gamma_N) = \frac{N}{k_B T(N-1)} \left[\hat{\mathbb{E}}[g(\Gamma)]\hat{\mathbb{E}}\left[\nabla_\theta U_\theta(\Gamma)\right]^\top - \hat{\mathbb{E}}\left[g(\Gamma)\cdot\nabla_\theta U_\theta(\Gamma)^\top\right]\right], \tag{4}$$

where $\hat{\mathbb{E}}[f(\Gamma)] = \frac{1}{N}\sum_{i=1}^N f(\Gamma_i)$ denotes an empirical mean over the samples.

This estimator provides an unbiased estimate of the Jacobian $\frac{\partial \mathbb{E}_{\Gamma\sim P_\theta(\Gamma)}[g(\Gamma)]}{\partial\theta}$. A proof is provided in Supplementary Section A.1. The estimator is related to the REINFORCE trick (Williams, 1992) and policy gradient estimators from reinforcement learning (Silver et al., 2014) when a Boltzmann state distribution is assumed. The result can also be derived using thermodynamic perturbation theory (Zwanzig, 1954; Thaler & Zavadlav, 2021).

Localized Boltzmann Estimator for Spatial Specificity. In some scenarios, unphysical configurations can occur within localized regions of the simulation domain, such as collisions between two molecules in a large condensed-phase system. Due to spatial averaging, global observables $g(\Gamma)$ may be insensitive to these localized events, limiting the ability to identify unphysical states. To address this, we propose the N-sample Localized Boltzmann Estimator. Here, the global energy $U_\theta(\Gamma)$ and observable $g(\Gamma)$ are replaced with local versions $U_\theta(\gamma)$ and $g(\gamma)$, where γ denotes a local neighborhood of $n<N$ atoms. Formally, define a neighborhood of n atoms $\mathcal{N} = \{x_1,x_2,\ldots,x_n \mid x_i \in \mathbb{Z}, 1 \le x_i \le N$ for all $i=1,2,\ldots,n\}$. The local state γ is defined as $\gamma = \{[r^{(\mathcal{N}_1)};\ldots;r^{(\mathcal{N}_n)}],[p^{(\mathcal{N}_1)};\ldots;p^{(\mathcal{N}_n)}]\}$. The local energy is then defined as $U_\theta(\gamma) = \sum_{i=1}^n U_\theta(\gamma^{(i)})$, where $\gamma^{(i)} = \{r^{(\mathcal{N}_i)},p^{(\mathcal{N}_i)}\}$ contains the position and momenta of the ith atom in the local neighborhood. The local energy is easily obtained by noting that MLFFs parameterize their global energy prediction U_θ as a sum over individual atomic energies. The localized estimator is thus given as follows:

Definition 2 (N-sample localized Boltzmann estimator)**.** *Given N i.i.d. samples of local states $\gamma_1,\ldots,\gamma_N$, where each γ_i is extracted from a global state $\Gamma_i \sim P_\theta(\Gamma)$, we define the N-sample localized estimator of the Jacobian $\frac{\partial \mathbb{E}_{\gamma\sim P_\theta(\gamma)}[g(\gamma)]}{\partial\theta}$ as*

$$\mathcal{E}(\gamma_1,\ldots,\gamma_N) \stackrel{def}{=} \frac{N}{k_B T(N-1)} \left[\hat{\mathbb{E}}[g(\gamma)]\hat{\mathbb{E}}[\nabla_\theta U_\theta(\gamma)]^\top - \hat{\mathbb{E}}\left[g(\gamma)\cdot\nabla_\theta U_\theta(\gamma)^\top\right]\right] \tag{5}$$

The localized estimator follows from the original Boltzmann estimator due to the fact that as a subset of the larger Boltzmann-distributed system, any local neighborhood also obeys a Boltzmann-distribution. In practice,

we extract multiple local neighborhoods from each global state to increase the sample size and state space coverage (Section A.6). A concrete example of localized instability, along with the use of our localized Boltzmann estimator to correct it, will be presented in the context of an all-atom water system in Results, Section 4.4.

Key Advantages. Unlike active learning approaches, no additional quantum mechanical energy and forces calculations are required to compute our Boltzmann Estimators (all energy terms are model predictions). As a result, learning with the Boltzmann estimator is computationally efficient. Due to the use of independent samples, the estimator also avoids numerical instability and memory demands associated with differentiating through long, continuous trajectories. Further, the most computationally expensive component of the estimator, the gradient of the potential energy $\nabla_\theta U_\theta(\Gamma)$, is independent of the observable $g(\Gamma)$. This means it can be reused, allowing efficient training to match multiple observables simultaneously.

3.2 Stability-Aware Boltzmann Estimator (StABlE) Training

Stability-Aware Boltzmann Estimator (StABlE) Training begins with conventional supervised pre-training of a MLFF on a reference dataset of energy and forces, using the loss defined in Equation 1. The method then proceeds by alternating between two major phases, simulation and learning, as illustrated in Figure 2.

Simulation Phase. The simulation phase aims to explore the molecular phase space and pinpoint regions where the MLFF becomes unstable. We sample R equilibrium states from the training dataset as initial conditions for separate MD trajectories (replicas). Using the pre-trained MLFF U_θ, we run MD simulations on these replicas in parallel for t timesteps. Due to our use of efficient vectorized GPU operations, simulating many replicas in parallel has similar computational cost to simulating a single replica. We apply a predetermined stability criterion (Supplementary Section A.5) to each replica, freezing those marked unstable at their current states. Simulations continue for the remaining replicas, with stability checks every t timesteps. Once a specified fraction of replicas become unstable, we rewind all unstable replicas by t timesteps. This ensures that further simulation for t timesteps will trigger instability, which is then corrected in the next phase. We note that many other strategies exist to explore phase space and find unstable regions, including Diffusion Monte Carlo with fictitious masses (Li et al., 2021) or uncertainty-based sampling (Kulichenko et al., 2023). StABlE Training can be flexibly used with any of these strategies without impacting the learning stage (described next) of the procedure.

Learning Phase. The learning phase aims to refine the MLFF to correct the previously encountered instabilities. Starting from the near-unstable configurations obtained in the simulation phase, we perform MD simulation for t timesteps, sampling every S^{th} state to obtain uncorrelated samples. Using the sampled states Γ_1,...,Γ_{N_d}, where $N_d = \frac{t}{S}$ $N_d = \frac{tR}{S}$, we compute the observable loss function (Equation 2). To update the MLFF parameters θ, we compute an unbiased estimator of the loss gradient via the Boltzmann Estimator (Section 3.1) and use it to perform a single step of gradient descent. We then reset all replicas to their original near-unstable states, simulate with the updated MLFF, recompute the loss and gradient estimator with the newly sampled states, and again update the MLFF weights. This process is repeated until the fraction of unstable replicas drops below a predetermined threshold. When this occurs, the learning phase has concluded, and we begin a new simulation phase starting from the endpoints of the last learning phase. We continue alternating between simulation and learning phases until a predetermined computational budget is reached, at which point StABlE Training has concluded. See Supplementary Section A.3 for a formal algorithmic description of the StABlE Training procedure.

Regularizing StABlE Training with Energy and Forces Reference Data. In practice, the mapping between a sparse set of system observables and a potential energy function is non-unique (Noid, 2013). Consequently, learning with observables alone is underconstrained. To combat this, we regularize the observable loss function (Equation 2) with the energy and forces loss function (Equation 1). The final StABlE loss function thus becomes,

$$\mathcal{L}_{\text{StABlE}}(\theta) \stackrel{\text{def}}{=} \mathcal{L}_{\text{obs}} + \lambda \mathcal{L}_{\text{QM}}, \tag{6}$$

where $\mathcal{L}_{\text{obs}}$ and $\mathcal{L}_{\text{QM}}$ were defined in Equations 2 and 1 respectively, and λ controls the strength of the regularization. Crucially, $\mathcal{L}_{\text{QM}}$ is only computed over the original training dataset $\mathcal{D}_{train}$, and not over new structures explored during MD simulation. Therefore, the regularization requires no additional *ab-initio* calculations.

4 Results

We present the results of StABlE Training on the aspirin molecule with SchNet (Section 4.1), Ac-Ala3-NHMe tetrapeptide with NequIP (Section 4.3), and an all-atom water system with GemNet-T (Section 4.4).

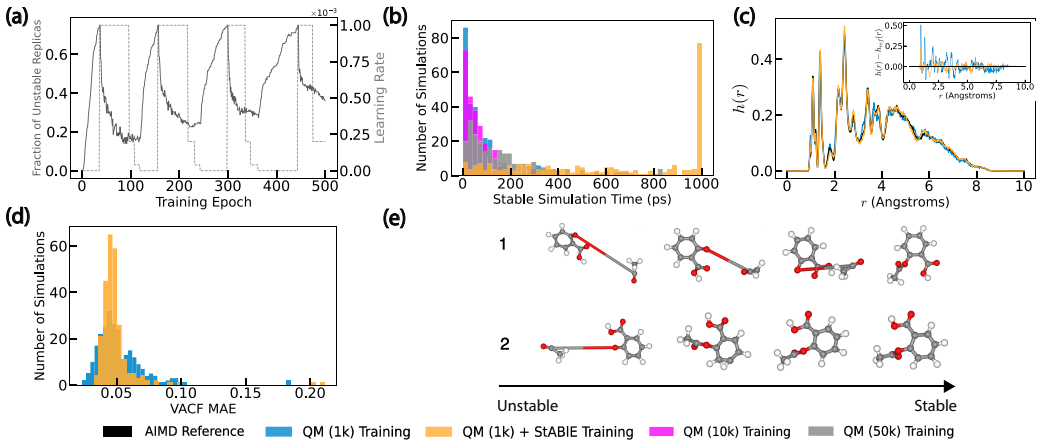

Figure 3: Aspirin simulation with StABlE Training. (a) Alternation of simulation and learning phases during StABlE Training with 128 parallel replicas. The simulation phases correspond to regions where the fraction of unstable replicas increases, while the learning phases correspond to regions where learning occurs and the fraction of unstable replicas decreases. **(b)** Stable simulation time of 256 parallel aspirin trajectories from SchNet MLFFs. Applying StABlE Training yields significantly more stable simulations than models trained only on energies and forces, surpassing models trained on 50× more QM data. **(c)** Distribution of interatomic distances ($h(r)$) from MLFF simulations. A StABlE-trained SchNet model closely recovers the true distribution of interatomic distances, while the model trained only on QM reference data produces a noisier $h(r)$ because it cannot stably simulate the system for longer time periods. Inset shows difference between predicted and reference $h(r)$. **(d)** Distribution of velocity autocorrelation function (VACF) mean absolute error (MAE). StABlE Training yields a reduction in variance across replicas. **(e)** Aspirin structures sampled over epochs of a single learning phase of StABlE Training. There is a clear progression as unstable configurations become stable.

4.1 Aspirin Molecule

Aspirin (chemical formula $C_9H_8O_4$) is the largest molecule from MD17 (Chmiela et al., 2017), a widely used benchmark dataset for atomistic simulations which contains energy and forces calculations computed at the PBE+vdW-TS (Tkatchenko & Scheffler, 2009) level of theory. Consisting of 21 atoms, aspirin has been shown to be the most challenging molecule in MD17 for state-of-the-art MLFFs to simulate stably (Fu et al., 2022). We pre-train a SchNet (Schütt et al., 2018) model on the energy and forces matching objective (Equation 1) using a subset of 1,000 aspirin structures from the reference dataset. After convergence, we begin StABlE Training with the global Boltzmann estimator, using 128 parallel replicas at a temperature of 500K. By simulating hundreds of replicas in parallel, we expose the MLFF to a comprehensive range of failure modes much more quickly than possible with a single replica. Following (Fu et al., 2022), we use a maximum bond length deviation criterion to detect instability in the simulations (Section A.5). We use the distribution of interatomic distances, $h(r)$, as our training observable, with the $h(r)$ computed over structures in the training dataset serving as the reference (Section A.4). This choice is motivated by the observation that unphysical bond stretches constitute the majority of failure cases in aspirin simulations. The $h(r)$ is sensitive to these abnormalities, and is thus an informative optimization criterion. We perform four complete StABlE cycles of simulation and learning with a SGD optimizer. We evaluate our final models by performing 1 nanosecond constant-temperature MD simulations, starting from 256 initial structures not seen during training. Figure 3a shows the progression of StABlE Training. During the simulation phase, the learning rate is zero and the fraction of unstable replicas steadily increases. When the upper instability threshold is reached, learning commences. The fraction of unstable replicas steadily decreases, confirming that optimization of the interatomic distance distribution produces the desired stability improvement.

A SchNet model trained via our stability-aware approach is significantly more stable during MD simulation than models trained only on the conventional QM energy and forces objective function (Figure 3b). As a result of our training procedure, the mean median stable simulation time increases from 42 to 602 picoseconds. Even when StABlE training is employed after training on only on 1,000 reference structures, it is more stable than models trained conventionally

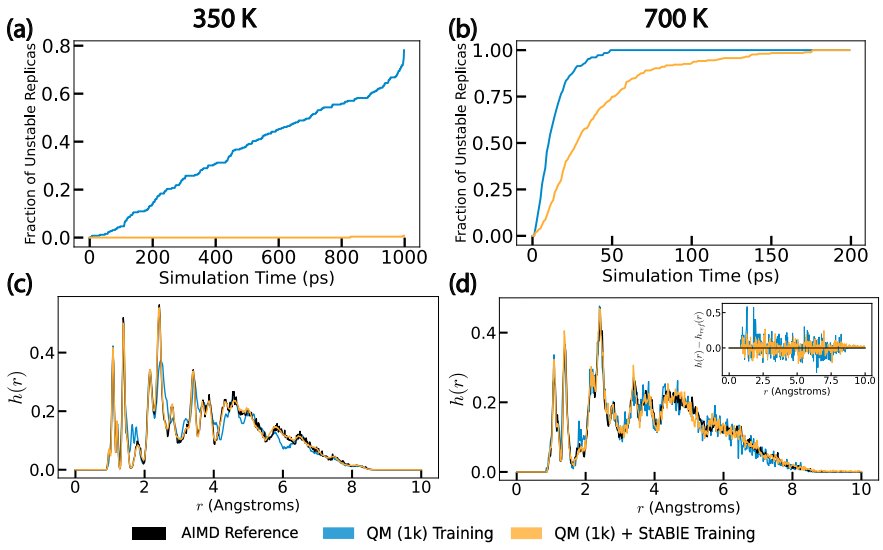

Figure 4: Testing temperature generalization for aspirin. (a, b) Fraction of unstable replicas as a function of simulation time for SchNet MLFFs at 350K and 700K. Applying StABlE Training yields significantly more stable simulations than a baseline SchNet model. **(c, d)** Distribution of interatomic distances ($h(r)$) from SchNet MLFFs. The StABlE-trained SchNet model more accurately recovers the true $h(r)$ at both 350K and 700K (the inset shows difference between the predicted and reference $h(r)$). The AIMD reference $h(r)$ is computed by Boltzmann-reweighting of independent samples from the 500K training dataset.

on 10,000 and 50,000 reference structures, highlighting the effectiveness of our method in improving stability without reliance on any additional QM reference data. We also observe in Supplementary Section A.12 that simply reducing the simulation timestep by a factor of ten does not eliminate instability in simulations produced by conventionally trained models, and reducing instability further may require impractically small timesteps. The stability gains from reducing the timestep are smaller than those resulting from StABlE Training, suggesting that our approach could enable taking larger timesteps and accelerate simulations. Simulations from our model closely recover the true distribution of interatomic distances, while the distributions produced by models solely trained on QM data are noisier, due to the limited stable simulation time (Figure 3c). We also test the ability of our StABlE-trained model to recover the velocity autocorrelation function (VACF), a fundamental dynamical observable not seen during training (Figure 3d). We see that the quality of the recovered VACF remains similar after StABlE Training, with lower variance over the simulated replicas (see Supplementary Section A.14 for a sample VACF computed from the simulations). The preservation of a held-out dynamical observable suggests that StABlE Training does not achieve stability improvements by simply restricting the model around a narrow regime corresponding to the reference $h(r)$. If this were the case, characteristic dynamic fluctuations around the reference $h(r)$ would be suppressed, leading to an inaccurate VACF. Figure 3e depicts aspirin structures sampled from simulations during the course of a single learning phase of StABlE Training, demonstrating that unphysical bond stretching in these structures is resolved by the training procedure.

4.2 Temperature generalization

We explore our method's generalization to different thermodynamic conditions. To do so, we perform 256 parallel MD simulations at 350K and 700K using the SchNet model which was trained with our StABlE Training procedure at 500K. We use the same criterion based on bond length deviation to detect instability as in Section 4.1. We estimate the reference $h(r)$ at 350K and 700K by Boltzmann-reweighting samples from the original 500K training dataset (Section A.9). Aspirin simulations produced by a StABlE-Trained SchNet model are significantly more stable at both temperatures than baseline models trained only on QM reference data (Figure 4a, b). At 350K, the difference is particularly large. After 1 nanosecond of simulation, less than 5% of virtually no replicas are unstable in the StABlE-trained MLFF simulation, while approximately 80% of replicas are unstable in the baseline MLFF simulation. The StABlE-trained model also recovers the true distribution of interatomic distances more closely than

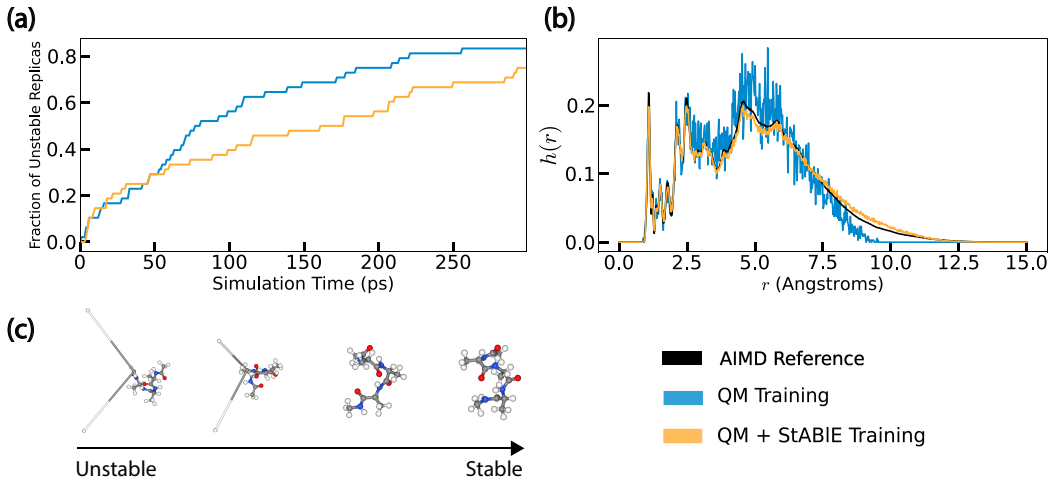

Figure 5: Ac-Ala3-NHMe tetrapeptide simulation with StABlE Training.(a) Fraction of unstable replicas as a function of simulation time for NequIP MLFFs. Applying StABlE Training yields a model which can simulate more MD replicas stably over time than a baseline trained only on energies and forces. **(b)** Distribution of interatomic distances ($h(r)$) from NequIP MLFF simulations. The StABlE-trained NequIP model much more closely recovers the true $h(r)$, while the $h(r)$ produced by the model trained only on QM reference data is noisy and inaccurate due to insufficient sampling time. **(c)** Ac-Ala3-NHMe structures sampled over epochs of a single learning phase of StABlE Training. There is a clear progression as very unstable configurations become stable.

the baseline MLFF at both temperatures (Figure 4c, d). At 350K, the baseline MLFF is stable enough produce a smooth $h(r)$, but the structure is inaccurate, indicating incorrect sampling of the phase space. At 700K, the baseline MLFF produces a $h(r)$ which is close to the AIMD-produced distribution, but is noisy due to insufficient stable simulation time. This underscores the necessity for a MLFF to be both stable and accurate in order to be practically useful in MD simulation. StABlE Training improves both of these metrics for this system. As generalization to new thermodynamic conditions is of crucial importance for MLFFs to be useful in practical applications (Kovács et al., 2021), we see the temperature-transferability of StABlE Training as an important strength of our method.

4.3 Ac-Ala3-NHMe Tetrapeptide

Ac-Ala3-NHMe (chemical formula $C_{12}H_{22}N_4O_4$) is a tetrapeptide from the MD22 dataset (Chmiela et al., 2022), a challenging benchmark consisting of reference energy and forces computed at the PBE+MBD (Perdew et al., 1996; Tkatchenko et al., 2012) level of theory for considerably larger molecules than those in the MD17 dataset. Relative to MD17 molecules, Ac-Ala3-NHMe poses unique challenges for atomistic simulations due to its larger size and flexibility (Kabylda et al., 2023). As a result, more expressive MLFFs—sometimes incorporating E(3) equivariance (Geiger & Smidt, 2022)—are required to accurately model the underlying potential energy surface. In our experiments, we use a NequIP (Batzner et al., 2022) model due to its promising accuracy and data efficiency on challenging atomistic simulation tasks (Kozinsky et al., 2023; Merchant et al., 2023). We pre-train a NequIP model on the QM energy and forces matching objective using 14,890 (or 25%) of available structures from the training dataset. After convergence, we begin StABlE Training with the global Boltzmann estimator, using 128 parallel replicas at a temperature of 500K. As with aspirin (Sections 4.1 - 4.2), we detect instability using the maximum bond length deviation criterion, and choose the distribution of interatomic distances, $h(r)$, as our training observable. The StABlE-trained NequIP model gives an improvement in stability relative to a baseline model trained only on QM reference data (Figure 5a). Additionally, the StABlE-trained model closely recovers the true $h(r)$, while the baseline MLFF produces a highly noisy $h(r)$. This is because of insufficient sampling, due to a lower stable simulation time (Figure 5b). The baseline MLFF also underestimates the latter portion of the $h(r)$ distribution, indicating poor modelling of long-range interactions. Figure 5c shows selected structures throughout the course of a single learning phase of StABlE Training, demonstrating a progression from very unstable to stable structures.

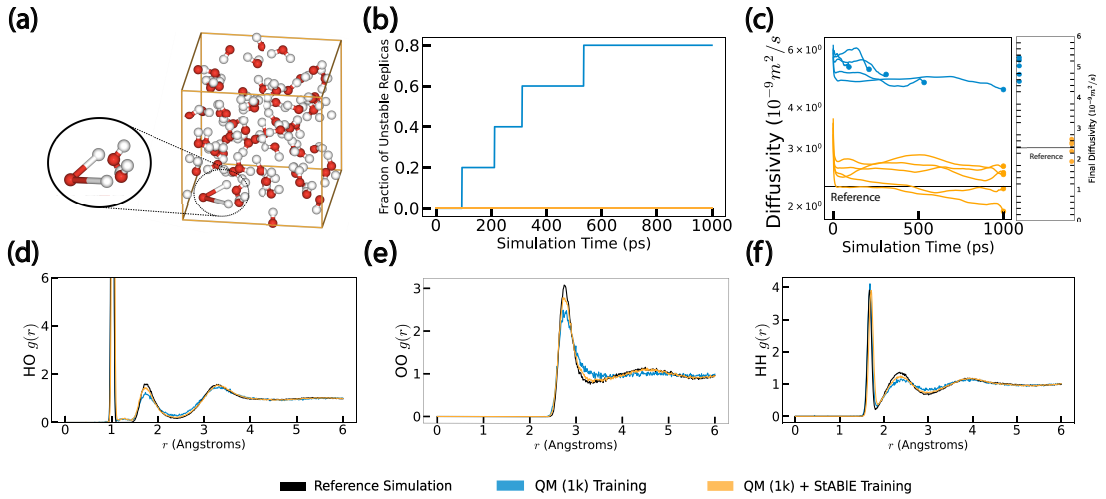

Figure 6: All-Atom Water Simulation with StABlE Training. (a) Example unstable configuration arising in a GemNet-T simulation of water. Instabilities are highly localized and manifest as unphysical bond stretches or intermolecular collisions, motivating the use of our localized Boltzmann estimator (Section 3.1) during training **(b)** Fraction of unstable replicas as a function of simulation time for GemNet-T MLFFs. StABlE Training yields a significant stability improvement relative to training on only energy and forces data. **(c)** Convergence of the diffusivity coefficient over MLFF simulations. The StABlE-trained model gives a considerably more accurate final estimate of the diffusivity coefficient. **(d - f)** Element-conditioned (OH, OO, HH) radial distribution functions (RDF) produced by MLFF simulations. The StABlE-trained model produces fewer unphysical collisions and more accurately captures long-range interactions.

4.4 Water

As a final evaluation of StABlE Training, we consider liquid-phase water. Water has historically posed unique challenges for atomistic simulations due to the presence of long-range interactions, proton disorder, and nuclear quantum effects (Cheng et al., 2019; Raza et al., 2011; Markland & Ceriotti, 2018). Additionally, estimating transport properties, such as the diffusivity coefficient, requires long, continuous trajectories to minimize statistical error. We use the dataset produced in Fu et al. (Fu et al., 2022), which contains reference all-atom simulations of 64 water molecules using the flexible Extended Simple Point Charge model (Wu et al., 2006) at 300K and 1 atm. For this system, we consider a GemNet-T model (Gasteiger et al., 2021). GemNet-T displays a failure mode in which unphysical configurations (e.g., bond stretching) first arise in highly localized "pockets" of 1-2 molecules (Figure 6a), which then gradually cascade to the rest of the simulation domain (Fu et al., 2022). Such configurations do not noticeably affect global system observables such as the radial distribution function. Therefore, in this setting it is more appropriate to use our localized version of the Boltzmann gradient estimator (Equation 5) in order to guide the optimization process towards specific spatial domains of instability. After pre-training a GemNet-T model using a subset of 1,000 reference structures, we begin StABlE Training with the localized Boltzmann estimator, simulating 5 parallel replicas at a temperature of 300K. We use a minimum intermolecular distance criterion (Section A.5) to detect instabilities. To estimate gradients for optimization, from each global state we extract all possible local neighborhoods containing a single water molecule (Supplementary Section A.6). The most prevalent failure mode produced by GemNet-T in this system is unphysical bond stretching, which eventually leads to unphysical coordination structures. Motivated by this observation, we use the mean hydrogen-oxygen bond length as our training observable. We perform a single cycle of simulation and learning. We then evaluate the performance of the StABlE-Trained model by performing 5 parallel nanosecond (ns) MD simulations at 300K, starting from held-out initial conditions. We achieve significant improvements in stability and accuracy using our training approach. We nearly double the stable simulation time relative to a baseline MLFF model trained only on QM reference data Our StABlE-trained model can simulate stably for 1 ns for all 5 initial conditions, while the baseline MLFF model trained only on QM reference data can only do so for 1 initial condition, and has a median stability of just 312 ps (Figure 6b). Compared to a reference value of $2.3 \times 10^{-9} \frac{m^2}{s}$, the mean diffusivity produced by StABlE-trained model

simulations is $1.6\times10^{-9}\frac{m^2}{s}$ $2.4\times10^{-9}\frac{m^2}{s}$, while the mean diffusivity produced by baseline model simulations is $5.0\times10^{-9}\frac{m^2}{s}$ $1.3\times10^{-9}\frac{m^2}{s}$ (Figure 6c). The long-range correlations in the element-conditioned RDFs (Figures 6d-f) are also captured more accurately by the StABlE-trained model. We reiterate that neither the element-conditioned RDFs nor the diffusivity coefficient were explicitly seen during StABlE Training. In Supplementary Section A.10, we demonstrate that StABlE Training leads to similar stability improvements when water simulations are conducted in the isothermal/isobaric (NPT) ensemble, instead of the canonical (NVT) ensemble.

5 Conclusion and Future Work

We have introduced StABlE Training, a strategy for training stable and accurate machine learning force fields. Our training procedure results in MLFFs which are significantly more stable in MD simulation, more accurately reproduce key simulation observables, (including those that were not explicitly trained on), exhibit better generalization to unseen temperatures, and have superior data efficiency relative to MLFFs trained only on QM data.

Key Takeaways. StABlE Training can utilize both quantum-mechanical energies and forces and system observables to supervise MLFF training. We highlight that the reference observables need not be acquired from experimental measurements, and can be computed by averaging over existing QM datasets. This makes our approach applicable in realistic computational discovery scenarios in which experimental characterization is not always available for hypothetical systems. The stability, accuracy, and data efficiency gains brought by StABlE Training require *no additional reference calculations or data*. The procedure is thus self-contained and efficient, requiring minimal additional computational expense beyond a single iteration of conventional MLFF training (see Supplementary Section A.6 for details on training times). This suggests that StABlE Training may scale more gracefully to larger MLFFs and atomistic systems than existing active learning approaches, which require repeated QM reference calculations and retraining. StABlE Training can also be flexibly applied across atomistic systems due to its ability to handle a diverse set of failure modes arising in MD simulation, including both global and local instabilities due to our localized Boltzmann estimator. This flexibility extends to the choice of MLFF: the effectiveness of StABlE Training across the three diverse architectures considered in this work suggests that our approach will remain applicable as MLFF architectures continue to evolve.

Limitations. Unlike traditional QM learning, training MLFFs to match reference observables lacks convergence guarantees in the large data limit. Specifically, the observable-matching objective is under-constrained, as the mapping between a potential energy function and a sparse set of simulation observables is in general non-unique (Noid, 2013). Although we mitigate this problem by pre-training and regularizing StABlE Training with the conventional energy and forces loss function, we observe that stability improvements resulting from StABlE Training are accompanied by a minor increase in the energy and forces error on a held-out test set, indicating a misalignment between the observable-matching and QM objectives. Adjusting the strength of the QM regularization and the learning rate used for StABlE Training are two primary ways in which to navigate this tradeoff. We note, however, that our observed error increases are typically within the range of DFT error (Supplementary Section A.11). We also note that StABlE Training is currently incompatible with dynamical observables, due to the use of uncorrelated states to compute the Boltzmann Estimator. Overcoming this limitation is non-trivial, as it requires optimization over long paths, which is a significantly higher-dimensional problem with largehigh-variance gradients (Bolhuis et al., 2023).

Future Work. In this work, we took observables calculated from high-fidelity simulations as reference values. Many of the observables we considered, such as the radial distribution function, diffusivity coefficient, and equilibrium bond length, are also experimentally measurable. Future work could explore training with experimental observables from multiple thermodynamic conditions simultaneously to yield more generalizable and robust potentials. Future work could also explore incorporating additional observables into StABlE Training, particularly those which are dynamical. This could restrict the set of learnable functions and address the under-constrained nature of learning with observables. We also note that the reference datasets used to pre-train MLFFs in this work were sampled uniformly from high-fidelity simulations. More sophisticated sampling strategies (Deringer et al., 2018; Karabin & Perez, 2020; Yoo et al., 2021; Fonseca et al., 2021; Kulichenko et al., 2023; Qi et al., 2024) for diverse dataset generation, as well as during phase space exploration to find unstable regions, could be employed along with StABlE Training to achieve further stability and accuracy gains. Closely related techniques such as active learning are compatible with StABlE Training, and could be used in tandem to further improve MLFF stability and robustness. In this setting, system observables can serve as a cheap source of information with which to augment the more expensive supervision provided by *ab-initio* calculations.

References

Brandon Amos and J. Zico Kolter. Optnet: Differentiable optimization as a layer in neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)*, pp. 136–145, 2017.

Ilyes Batatia, David P. Kovacs, Gregor Simm, Christoph Ortner, and Gabor Csanyi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. In *Neural Information Processing Systems (NeurIPS)*, 2022.

Ilyes Batatia, Philipp Benner, Yuan Chiang, Alin M. Elena, Dávid P. Kovács, Janosh Riebesell, Xavier R. Advincula, Mark Asta, William J. Baldwin, Noam Bernstein, Arghya Bhowmik, Samuel M. Blau, Vlad Cărare, James P. Darby, Sandip De, Flaviano Della Pia, Volker L. Deringer, Rokas Elijošius, Zakariya El-Machachi, Edvin Fako, Andrea C. Ferrari, Annalena Genreith-Schriever, Janine George, Rhys E. A. Goodall, Clare P. Grey, Shuang Han, Will Handley, Hendrik H. Heenen, Kersti Hermansson, Christian Holm, Jad Jaafar, Stephan Hofmann, Konstantin S. Jakob, Hyunwook Jung, Venkat Kapil, Aaron D. Kaplan, Nima Karimitari, Namu Kroupa, Jolla Kullgren, Matthew C. Kuner, Domantas Kuryla, Guoda Liepuoniute, Johannes T. Margraf, Ioan-Bogdan Magdău, Angelos Michaelides, J. Harry Moore, Aakash A. Naik, Samuel P. Niblett, Sam Walton Norwood, Niamh O'Neill, Christoph Ortner, Kristin A. Persson, Karsten Reuter, Andrew S. Rosen, Lars L. Schaaf, Christoph Schran, Eric Sivonxay, Tamás K. Stenczel, Viktor Svahn, Christopher Sutton, Cas van der Oord, Eszter Varga-Umbrich, Tejs Vegge, Martin Vondrák, Yangshuai Wang, William C. Witt, Fabian Zills, and Gábor Csányi. A foundation model for atomistic materials chemistry, 2023.

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P. Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E. Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature Communications*, 13(1), 5 2022. doi: 10.1038/s41467-022-29939-5.

Vaibhav Bihani, Utkarsh Pratiush, Sajid Mannan, Tao Du, Zhimin Chen, Santiago Miret, Matthieu Micoulaut, Morten M Smedskjaer, Sayan Ranu, and N M Anoop Krishnan. Egraffbench: Evaluation of equivariant graph neural network force fields for atomistic simulations, 2023.

Mathieu Blondel, Quentin Berthet, Marco Cuturi, Roy Frostig, Stephan Hoyer, Felipe Llinares-López, Fabian Pedregosa, and Jean-Philippe Vert. Efficient and modular implicit differentiation. In *Neural Information Processing Systems (NeurIPS)*. Google Research, 2022.

Peter G. Bolhuis, Z. Faidon Brotzakis, and Bettina G. Keller. Optimizing molecular potential models by imposing kinetic constraints with path reweighting. *J. Chem. Phys.*, 159:074102, 2023. doi: 10.1063/5.0151166. URL `https://doi.org/10.1063/5.0151166`.

Giovanni Bussi and Michele Parrinello. Accurate sampling using langevin dynamics. *Physical Review E—Statistical, Nonlinear, and Soft Matter Physics*, 75(5):056707, 2007.

R. Car and M. Parrinello. Unified approach for molecular dynamics and density-functional theory. *Phys. Rev. Lett.*, 55:2471, 1985. doi: 10.1103/PhysRevLett.55.2471. Published 25 November 1985.

Scott P. Carmichael and M. Scott Shell. A new multiscale algorithm and its application to coarse-grained peptide models for self-assembly. *J. Phys. Chem. B*, 116(29):8383–8393, 2012. doi: 10.1021/jp2114994.

Nicholas E. Charron, Felix Musil, Andrea Guljas, Yaoyi Chen, Klara Bonneau, Aldo S. Pasos-Trejo, Jacopo Venturin, Daria Gusew, Iryna Zaporozhets, Andreas Krämer, Clark Templeton, Atharva Kelkar, Aleksander E. P. Durumeric, Simon Olsson, Adrià Pérez, Maciej Majewski, Brooke E. Husic, Ankit Patel, Gianni De Fabritiis, Frank Noé, and Cecilia Clementi. Navigating protein landscapes with a machine-learned transferable coarse-grained model, 2023.

Anthony K Cheetham and Ram Seshadri. Artificial intelligence driving materials discovery? perspective on the article: Scaling deep learning for materials discovery. *Chemistry of Materials*, 36(8):3490–3495, 2024.

C. Chen and S.P. Ong. A universal graph deep learning interatomic potential for the periodic table. *Nat Comput Sci*, 2:718–728, 2022. doi: 10.1038/s43588-022-00349-3. URL `https://doi.org/10.1038/s43588-022-00349-3`.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations, 2019.

Bingqing Cheng, Edgar A. Engel, Jörg Behler, and Michele Ceriotti. Ab initio thermodynamics of liquid and solid water. *Proc. Natl. Acad. Sci. U.S.A.*, 116(4):1110–1115, 1 2019. doi: 10.1073/pnas.1815117116. URL `https://doi.org/10.1073/pnas.1815117116`. Edited by Pablo G. Debenedetti, Princeton University, Princeton, NJ, approved December 3, 2018 (received for review September 4, 2018).

Stefan Chmiela, Alexandre Tkatchenko, Huziel E. Sauceda, Igor Poltavsky, Kristof T. Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5), 5 2017. doi: 10.1126/sciadv.1603015.

Stefan Chmiela, Valentin Vassilev-Galindo, Oliver T. Unke, Adil Kabylda, Huziel E. Sauceda, Alexandre Tkatchenko, and Klaus-Robert Müller. Accurate global machine learning force fields for molecules with hundreds of atoms, 2022.

Wendy D. Cornell, Piotr Cieplak, Christopher I. Bayly, Ian R. Gould, Kenneth M. Merz, David M. Ferguson, David C. Spellmeyer, Thomas Fox, James W. Caldwell, and Peter A. Kollman. A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. *J. Am. Chem. Soc.*, 117:5179–5197, 1995.

Bowen Deng, Peichen Zhong, KyuJung Jun, Janosh Riebesell, Kevin Han, Christopher J Bartel, and Gerbrand Ceder. Chgnet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence*, 5(9):1031–1041, 2023.

Volker L Deringer, Chris J Pickard, and Gábor Csányi. Data-driven learning of total and local energies in elemental boron. *Physical review letters*, 120(15):156001, 2018.

FA Faber, L Hutchison, B Huang, J Gilmer, SS Schoenholz, GE Dahl, O Vinyals, S Kearnes, PF Riley, and OA von Lilienfeld. Prediction errors of molecular machine learning models lower than hybrid dft error. *J Chem Theory Comput*, 13(11):5255–5264, 11 2017. doi: 10.1021/acs.jctc.7b00577. Epub 2017 Oct 10.

Gregory Fonseca, Igor Poltavsky, Valentin Vassilev-Galindo, and Alexandre Tkatchenko. Improving molecular force fields across configurational space by combining supervised and unsupervised machine learning. *The Journal of Chemical Physics*, 154(12), 2021.

Daan Frenkel and Berend Smit. *Understanding Molecular Simulation: From Algorithms to Applications*. Academic Press, 2 edition, 11 2001.

Richard A. Friesner. Ab initio quantum chemistry: Methodology and applications. *Proceedings of the National Academy of Sciences*, 102(19):6648–6653, 5 2005. doi: 10.1073/pnas.0408036102.

Xiang Fu, Zhenghao Wu, Wujie Wang, Tian Xie, Sinan Keten, Rafael Gomez-Bombarelli, and Tommi Jaakkola. Forces are not enough: Benchmark and critical evaluation for machine learning force fields with molecular simulations, 2022.

Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. In *8th International Conference on Learning Representations, ICLR*, 2020.

Johannes Gasteiger, Florian Becker, and Stephan Günnemann. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2021.

Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ulissi, C. Lawrence Zitnick, and Abhishek Das. Gemnet-oc: Developing graph neural networks for large and diverse molecular simulation datasets, 2022.

Fuchun Ge, Ran Wang, Chen Qu, Peikun Zheng, Apurba Nandi, Riccardo Conte, Paul L. Houston, Joel M. Bowman, and Pavlo O. Dral. Tell machine learning potentials what they are needed for: Simulation-oriented training exemplified for glycine. *J. Phys. Chem. Lett.*, 15(16):4451–4460, 2024. doi: 10.1021/acs.jpclett.4c00746. URL `https://doi.org/10.1021/acs.jpclett.4c00746`. Publication Date: April 16, 2024.

Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks, 2022.

S. Gould, B. Fernando, A. Cherian, P. Anderson, R. S. Cruz, and E. Guo. On differentiating parameterized argmin and argmax problems with application to bi-level optimization. *arXiv preprint arXiv:1607.05447*, 2016.

Weihua Hu, Muhammed Shuaibi, Abhishek Das, Siddharth Goyal, Anuroop Sriram, Jure Leskovec, Devi Parikh, and C. Lawrence Zitnick. Forcenet: A graph neural network for large-scale quantum calculations, 2021.

John Ingraham, Adam Riesselman, Chris Sander, and Debora Marks. Learning protein structure with a differentiable simulator. In *International Conference on Learning Representations, ICLR*, 2019.

A. Kabylda, V. Vassilev-Galindo, S. Chmiela, I. Poltavsky, and A. Tkatchenko. Efficient interatomic descriptors for accurate machine learning force fields of extended molecules. *Nat Commun*, 14(1):3562, 6 2023. doi: 10.1038/s41467-023-39214-w. Erratum in: Nat Commun. 2023 Jul 11;14(1):4116. PMID: 37322039; PMCID: PMC10272221.

Mariia Karabin and Danny Perez. An entropy-maximization approach to automated training set generation for interatomic potentials. *The Journal of Chemical Physics*, 153(9), 2020.

Christopher Kolloff and Simon Olsson. Rescuing off-equilibrium simulation data through dynamic experimental data with dynammo. *Mach. Learn.: Sci. Technol.*, 4(4):045050, 2023. doi: 10.1088/2632-2153/ac7b4f. © 2023 The Author(s).

Dávid Péter Kovács, Cas van der Oord, Jiri Kucera, Alice E. A. Allen, Daniel J. Cole, Christoph Ortner, and Gábor Csányi. Linear atomic cluster expansion force fields for organic molecules: Beyond rmse. *Journal of Chemical Theory and Computation*, 17(12):7696–7711, 2021. doi: 10.1021/acs.jctc.1c00647. URL `https://doi.org/10.1021/acs.jctc.1c00647`. PMID: 34735161.

Dávid Péter Kovács, J. Harry Moore, Nicholas J. Browning, Ilyes Batatia, Joshua T. Horton, Venkat Kapil, William C. Witt, Ioan-Bogdan Magdău, Daniel J. Cole, and Gábor Csányi. Mace-off23: Transferable machine learning force fields for organic molecules, 2023.

Boris Kozinsky, Albert Musaelian, Anders Johansson, and Simon Batzner. Scaling the leading accuracy of deep equivariant models to biomolecular simulations of realistic size. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '23, New York, NY, USA, 2023. Association for Computing Machinery. doi: 10.1145/3581784.3627041. URL `https://doi.org/10.1145/3581784.3627041`.

Aditi S Krishnapriyan, Alejandro F Queiruga, N Benjamin Erichson, and Michael W Mahoney. Learning continuous models for continuous physics. *Communications Physics*, 6(1):319, 2023.

M. Kulichenko, K. Barros, N. Lubbers, et al. Uncertainty-driven dynamics for active learning of interatomic potentials. *Nat Comput Sci*, 3:230–239, 2023. doi: 10.1038/s43588-023-00406-5. URL `https://doi.org/10.1038/s43588-023-00406-5`.

H. G. Kümmel. A biography of the coupled cluster method. In R. F. Bishop, T. Brandes, K. A. Gernoth, N. R. Walet, and Y. Xian (eds.), *Recent Progress in Many-Body Theories*, pp. 334–348. World Scientific Publishing, Singapore, 2002.

Ask Hjorth Larsen, Jens Jørgen Mortensen, Jakob Blomqvist, Ivano E. Castelli, Rune Christensen, Marcin Dulak, Jesper Friis, Michael N. Groves, Bjørk Hammer, Cory Hargus, et al. The atomic simulation environment — a python library for working with atoms. *Journal of Physics: Condensed Matter*, 2017. doi: 10.1088/1361-648X/aa680e.

J. Li, C. Qu, and J. M. Bowman. Diffusion monte carlo with fictitious masses finds holes in potential energy surfaces. *Molecular Physics*, 119(17-18), 2021. doi: 10.1080/00268976.2021.1976426. URL `https://doi.org/10.1080/00268976.2021.1976426`.

Xiang-Guo Li, Chongze Hu, Chi Chen, Zhi Deng, Jian Luo, and Shyue Ping Ong. Quantum-accurate spectral neighbor analysis potential models for ni-mo binary alloys and fcc metals. *Phys. Rev. B*, 98:094104, 2018. doi: 10.1103/PhysRevB.98.094104. URL `https://link.aps.org/doi/10.1103/PhysRevB.98.094104`.

Qidong Lin, Liang Zhang, Yaolong Zhang, and Bin Jiang. Searching configurations in uncertainty space: Active learning of high-dimensional neural network reactive potentials. *J. Chem. Theory Comput.*, 17(5):2691–2701, 2021. doi: 10.1021/acs.jctc.0c01193.

Yi Liu, Limei Wang, Meng Liu, Yuchao Lin, Xuan Zhang, Bora Oztekin, and Shuiwang Ji. Spherical message passing for 3d molecular graphs. In *International Conference on Learning Representations (ICLR)*, 2022.

M. Majewski, A. Pérez, P. Thölke, et al. Machine learning coarse-grained potentials of protein thermodynamics. *Nat Commun*, 14:5739, 2023. doi: 10.1038/s41467-023-41343-1. URL https://doi.org/10.1038/s41467-023-41343-1.

TE Markland and M Ceriotti. Nuclear quantum effects enter the mainstream. *Nat Rev Chem*, 2:0109, 2018. doi: 10.1038/s41570-017-0109.

S. J. Marrink, H. J. Risselada, S. Yefimov, D. P. Tieleman, and A. H. De Vries. The martini force field: coarse grained model for biomolecular simulations. *J. Phys. Chem. B*, 111:7812–7824, 2007. doi: 10.1021/jp071097f.

A. Merchant, S. Batzner, S.S. Schoenholz, et al. Scaling deep learning for materials discovery. *Nature*, 624:80–85, 2023. doi: 10.1038/s41586-023-06735-9.

Joe D. Morrow, John L. A. Gardner, and Volker L. Deringer. How to validate machine-learned interatomic potentials. *J. Chem. Phys.*, 158:121501, 2023.

Albert Musaelian, Simon Batzner, Anders Johansson, Lixin Sun, Cameron J. Owen, Mordechai Kornbluth, and Boris Kozinsky. Learning local equivariant representations for large-scale atomistic dynamics, 2022.

Carles Navarro, Maciej Majewski, and Gianni De Fabritiis. Top-down machine learning of coarse-grained protein force fields. *J. Chem. Theory Comput.*, 2023.

Geoffrey Négiar, Michael W. Mahoney, and Aditi Krishnapriyan. Learning differentiable solvers for systems with hard constraints. In *The Eleventh International Conference on Learning Representations*, 2023.

W. G. Noid. Perspective: coarse-grained models for biomolecular systems. *J. Chem. Phys.*, 139:090901, 2013.

John P. Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Phys. Rev. Lett.*, 77:3865–3868, 10 1996. doi: 10.1103/PhysRevLett.77.3865. URL https://link.aps.org/doi/10.1103/PhysRevLett.77.3865.

Ji Qi, Tsz Wai Ko, Brandon C Wood, Tuan Anh Pham, and Shyue Ping Ong. Robust training of machine learning interatomic potentials with dimensionality reduction and stratified sampling. *npj Computational Materials*, 10(1):43, 2024.

Z Raza et al. Proton ordering in cubic ice and hexagonal ice; a potential new ice phase–xic. *Phys. Chem. Chem. Phys.*, 13:19788–19795, 2011. doi: 10.1039/c1cp22167a.

Jie Ren, Xidong Feng, Bo Liu, Xuehai Pan, Yao Fu, Luo Mai, and Yaodong Yang. Torchopt: An efficient library for differentiable optimization, 2022.

Sebastien Röcken and Julija Zavadlav. Accurate machine learning force fields via experimental and simulation data fusion, 2023.

Michael Schaarschmidt, Morgane Riviere, Alex M. Ganose, James S. Spencer, Alexander L. Gaunt, James Kirkpatrick, Simon Axelrod, Peter W. Battaglia, and Jonathan Godwin. Learned force fields are ready for ground state catalyst discovery, 2022.

Christoph Schran, Krystof Brezina, and Ondrej Marsalek. Committee neural network potentials control generalization errors and enable active learning. *J. Chem. Phys.*, 153(10):104105, Sep 2020. doi: 10.1063/5.0016004. URL https://doi.org/10.1063/5.0016004.

K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller. Schnet – a deep learning architecture for molecules and materials. *J. Chem. Phys.*, 148(24):241722, 6 2018. doi: 10.1063/1.5019779. URL https://doi.org/10.1063/1.5019779.

Kristof T. Schütt, Oliver T. Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning (ICML)*, 2021.

Nima Shoghi, Adeesh Kolluru, John R. Kitchin, Zachary W. Ulissi, C. Lawrence Zitnick, and Brandon M. Wood. From molecules to materials: Pre-training large generalizable models for atomic property prediction, 2023.

David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *International conference on machine learning*, pp. 387–395. Pmlr, 2014.

Justin S. Smith, Ben Nebgen, Nicholas Lubbers, Olexandr Isayev, and Adrian E. Roitberg. Less is more: Sampling chemical space with active learning. *J. Chem. Phys.*, 148:241733, 2018. doi: 10.1063/1.5023802.

Sina Stocker, Johannes Gasteiger, Florian Becker, Stephan Günnemann, and Johannes T Margraf. How robust are modern graph neural network potentials in long and hot molecular dynamics simulations? *Machine Learning: Science and Technology*, 3(4):045010, 11 2022. doi: 10.1088/2632-2153/ac9955. URL `https://dx.doi.org/10.1088/2632-2153/ac9955`.

S. Takamoto, C. Shinagawa, D. Motoki, et al. Towards universal neural network potential for material discovery applicable to arbitrary combination of 45 elements. *Nat Commun*, 13:2991, 2022. doi: 10.1038/s41467-022-30687-9. URL `https://doi.org/10.1038/s41467-022-30687-9`.

Sebastian Thaler and Julija Zavadlav. Learning neural network potentials from experimental data via differentiable trajectory reweighting. *Nature Communications*, 12:6884, 2021. doi: 10.1038/s41467-021-27241-4.

Alexandre Tkatchenko and Matthias Scheffler. Accurate molecular van der waals interactions from ground-state electron density and free-atom reference data. *Phys. Rev. Lett.*, 102:073005, 2 2009. doi: 10.1103/PhysRevLett.102.073005. URL `https://link.aps.org/doi/10.1103/PhysRevLett.102.073005`.

Alexandre Tkatchenko, Robert A. DiStasio, Roberto Car, and Matthias Scheffler. Accurate and efficient method for many-body van der waals interactions. *Phys. Rev. Lett.*, 108:236402, 6 2012. doi: 10.1103/PhysRevLett.108.236402. URL `https://link.aps.org/doi/10.1103/PhysRevLett.108.236402`.

J. Vandermause, S.B. Torrisi, S. Batzner, et al. On-the-fly active learning of interpretable bayesian force fields for atomistic rare events. *npj Comput Mater*, 6:20, 2020. doi: 10.1038/s41524-020-0283-z. URL `https://doi.org/10.1038/s41524-020-0283-z`.

Joshua A. Vita and Daniel Schwalbe-Koda. Data efficiency and extrapolation trends in neural network interatomic potentials. *Mach. Learn.: Sci. Technol.*, 4:035031, 2023.

Wujie Wang, Simon Axelrod, and Rafael Gómez-Bombarelli. Differentiable molecular simulations for control and learning, 2020.

Wujie Wang, Zhenghao Wu, Johannes C. B. Dietschreit, and Rafael Gó mez-Bombarelli. Learning pair potentials using differentiable simulations. *The Journal of Chemical Physics*, 158(4):044113, 1 2023a. doi: 10.1063/5.0126475.

Zun Wang, Hongfei Wu, Lixin Sun, Xinheng He, Zhirong Liu, Bin Shao, Tong Wang, and Tie-Yan Liu. Improving machine learning force fields for molecular dynamics simulations with fine-grained force metrics. *J. Chem. Phys.*, 159:035101, 2023b. doi: 10.1063/5.0147023.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 1992.

Yujie Wu, Harald L. Tepper, and Gregory A. Voth. Flexible simple point-charge water model with improved liquid-state properties. *The Journal of Chemical Physics*, 124(2):024503, 1 2006. ISSN 0021-9606, 1089-7690. doi: 10.1063/1.2136877.

Dongsun Yoo, Jisu Jung, Wonseok Jeong, and Seungwu Han. Metadynamics sampling in atomic environment space for collecting training data for machine learning potentials. *npj Computational Materials*, 7(1):131, 2021.

Robert W. Zwanzig. High-temperature equation of state by a perturbation method. i. nonpolar gases. *J. Chem. Phys.*, 22:1420–1426, 1954.

Martin Šípka, Johannes C. B. Dietschreit, Lukáš Grajciar, and Rafael Gómez-Bombarelli. Differentiable simulations for enhanced sampling of rare events, 2023.