



PAPER • OPEN ACCESS

## Projected Hessian learning: fast curvature supervision for accurate machine-learning interatomic potentials

To cite this article: Austin Rodriguez *et al* 2026 *Mach. Learn.: Sci. Technol.* **7** 055037

View the [article online](#) for updates and enhancements.

### You may also like

- [Efficient generation of stable linear machine-learning force fields with uncertainty-aware active learning](#)  
Valerio Briganti and Alessandro Lunghi
- [Exploration of transferable and uniformly accurate neural network interatomic potentials using optimal experimental design](#)  
Viktor Zaverkin and Johannes Kästner
- [Gradient guided furthest point sampling for robust training set selection](#)  
Morris Trestman, Stefan Gugler, Felix A Faber et al.



## PAPER

## OPEN ACCESS

## RECEIVED

4 March 2026

## REVISED

28 August 2026

## ACCEPTED FOR PUBLICATION

25 September 2026

## PUBLISHED

9 October 2026

Original content from this work may be used under the terms of the [Creative Commons Attribution 4.0 licence](#).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.



# Projected Hessian learning: fast curvature supervision for accurate machine-learning interatomic potentials

Austin Rodriguez<sup>1,2</sup> , Justin S Smith<sup>3</sup> , Sakib Matin<sup>1</sup> , Nicholas Lubbers<sup>4,\*</sup> , Kipton Barros<sup>1,5,\*</sup> and Jose L Mendoza-Cortes<sup>2,6,\*</sup>

<sup>1</sup> Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, United States of America

<sup>2</sup> Department of Chemical Engineering & Materials Science, Michigan State University, East Lansing, MI 48824, United States of America

<sup>3</sup> NVIDIA Corp., 2788 San Tomas Expy, Santa Clara, CA 95051, United States of America

<sup>4</sup> Computer and Artificial Intelligence Division, Los Alamos National Laboratory, Los Alamos, NM 87545, United States of America

<sup>5</sup> Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM 87545, United States of America

<sup>6</sup> Department of Physics and Astronomy, Michigan State University, East Lansing, MI 48823, United States of America

\* Authors to whom any correspondence should be addressed.

E-mail: [nlubbers@lanl.gov](mailto:nlubbers@lanl.gov), [kbarros@lanl.gov](mailto:kbarros@lanl.gov) and [jmendoza@msu.edu](mailto:jmendoza@msu.edu)

**Keywords:** machine-learning interatomic potentials (MLIPs), Hessian–vector products (HVPs), second-derivative training, stochastic trace estimation, Hutchinson estimator, transition states and reaction pathways, data-efficient training

Supplementary material for this article is available [online](#)

## Abstract

The Hessian matrix of second derivatives contains substantially richer information about the local geometry of the potential energy surface than energies and forces alone. Although incorporating full Hessians into machine-learning interatomic potential (MLIP) training can significantly improve accuracy and robustness, the quadratic computational and memory cost of explicitly constructing and storing Hessian matrices has limited its practical use. Here, we introduce *Projected Hessian Learning* (PHL), a scalable second-order training framework that incorporates curvature information using only Hessian–vector products (HVPs). By avoiding explicit Hessian construction and instead projecting curvature along stochastic probe directions, PHL reduces the cost of second-derivative supervision to near force-level complexity. The resulting dense-probe trace estimator is unbiased and exhibits favorable scaling with system size, enabling curvature-informed training without quadratic memory growth. We evaluated various training approaches on a chemically diverse dataset of reactants, products, transition states, intrinsic reaction coordinates, and normal-mode sampled geometries generated at the theoretical level  $\omega$ B97XD/6-31G(d). Four training schemes are considered: energy-force training (E-F), two HVP-based approaches (E-F-HVP with one-hot or dense Gaussian vectors) and energy-force-Hessian training (E-F-H). When probe vectors are randomized for each minibatch, both HVP-based approaches achieve energy, force, and Hessian accuracy comparable to full Hessian training while providing more than  $24\times$  speedups per epoch for the small molecular systems studied here. In a more realistic fixed-vector regime with only one HVP per molecule, dense Gaussian projections consistently outperform one-column probing, particularly for far-from-equilibrium geometries. Paired follow-up training with Rademacher probes gives no statistically significant change in model accuracy relative to the original Gaussian choice, although Rademacher probes have lower single-probe estimator variance and bounded component magnitudes. Overall, PHL effectively replaces explicit Hessian supervision with force-complexity curvature training, retaining most of the accuracy benefits of full second-order methods while enabling scalable MLIP development for larger and more complex molecular systems.

## 1. Introduction

The quality of training data often imposes a fundamental limit on the accuracy of a machine learning interatomic potential (MLIP). [1] In neural-network-based MLIPs, reference data are typically obtained from quantum chemistry calculations such as density functional theory (DFT) or coupled cluster, and the ML model is trained to reproduce these reference energies while enabling orders-of-magnitude faster inference [1]. A key insight established across multiple MLIP frameworks is that including gradients of the energy with respect to atomic coordinates (i.e. atomic forces) in the training loss substantially improves both accuracy and generalization. This paradigm has been realized in a wide range of architectures, including Behler–Parrinello neural networks [2], ANI models [3–6], Hessian interatomic potentials (HIP)-NN [7], the DeepMD framework [8], kernel/descriptor-based MLIPs such as Gaussian approximation potentials, [9] spectral neighbor analysis potentials, [10] and moment tensor potentials, [11] as well as message-passing approaches such as SchNet and NequIP [12, 13]. More recent equivariant message-passing architectures, e.g. PaiNN and MACE, [14, 15] further reinforce the central role of force supervision in improving data efficiency and robustness. In all of these works, joint training on energies and forces consistently yields lower prediction errors and improved robustness compared to training on energy alone [16, 17], reflecting the strong regularizing effect and the enhanced efficiency of the data provided by the first-derivative information. While energy and force training has become standard, response properties that depend explicitly on second derivatives can remain inaccurate even when energies and forces are well reproduced. In crystalline materials, for example, harmonic phonons are determined by the (mass-weighted) force-constant matrix, i.e. the energy Hessian, and recent benchmarks show that several high-performing universal MLIPs still exhibit substantial errors in phonon observables despite strong energy and force accuracy [18].

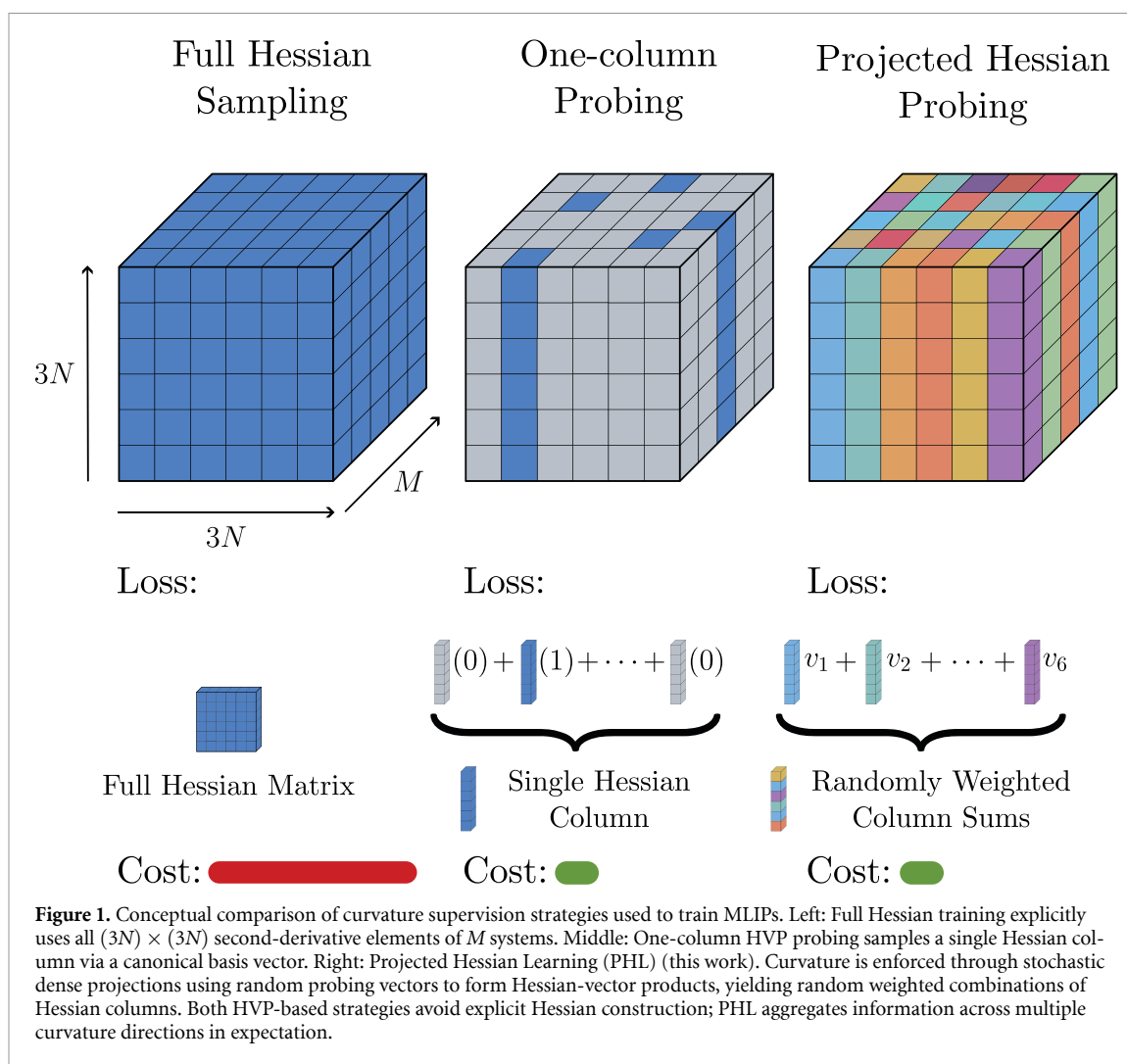
A well-implemented quantum chemistry code can produce not just the total energy but also the gradient of energy with respect to atom positions. Indeed, the principle of reverse-mode automatic differentiation [19, 20], known as backpropagation in machine learning, ensures that all force components can be calculated at a cost comparable to that of the energy evaluation alone; in electronic structure theory this is closely related to analytic energy-derivative techniques used to obtain nuclear forces [21]. Consequently, it is now standard for DFT codes to provide forces in addition to energies. The situation is different, however, for the Hessian matrix of second derivatives,

$$H_{(i,\alpha),(j,\beta)} = \frac{\partial^2 E}{\partial r_i^\alpha \partial r_j^\beta}, \quad (1)$$

where  $r_i^{\{x,y,z\}}$  denotes components of the position of the atom  $i$ . Elements of the Hessian can be understood as the sensitivity of the force on an atom  $i$  with respect to perturbations to the position of an atom  $j$ . Calculating all such sensitivities is substantially slower than calculating forces alone, as analytic second derivatives generally require additional response equations beyond those used for gradients [21, 22].

For a system of  $N$  atoms, the total energy is a single scalar, whereas the full gradient is a  $3N$ -dimensional vector, dramatically expanding the informational content available for training. This naturally raises the question of whether second derivatives of the energy, i.e. curvature information, can further improve the accuracy and generalization of the MLIP. The second derivative matrix, or Hessian, contains  $(3N)^2$  components and governs the local geometry of the potential energy surface (PES), including vibrational frequencies, transition-state curvature, and reaction pathways [21, 23, 24]. Motivated by these physical insights, recent work has begun to explore Hessian-informed approaches in ML potential development, including direct Hessian learning [25], analytical ML Hessians for the characterization of transition-states [26], Hessian distillation frameworks for specialized force fields [27], and the use of Hessians as diagnostic tools to assess force locality in MLIPs [28]. Very recently, Koker *et al* proposed phonon fine-tuning (PFT), which directly supervises second-order force constants by matching MLIP energy Hessians to DFT-derived force constants from finite-displacement phonon workflows [29]. In a complementary direction, Burger *et al* introduced HIP, which predict symmetry-preserving Hessians directly from equivariant message-passing features without relying on automatic differentiation or finite differences [30].

Complementing these methodological developments, the availability of *curvature-labeled* datasets is rapidly expanding. For equilibrium molecular structures, *Hessian QM9* provides numerical DFT Hessians for 41 645 QM9 molecules (including vacuum and implicit-solvent environments), enabling systematic evaluation of Hessian-aware training for vibrational properties [31]. For reactive chemistry, the recently released *HORM* database scales this idea dramatically, reporting 1.84 million quantum-chemistry



Hessian matrices for diverse reactive geometries and transition-state workflows [32]. The dataset used in this work complements these resources by targeting chemically diverse reaction pathways with equilibrium reactants, products, and transition states (RTP), continuous intrinsic reaction coordinate trajectories (IRC), and far-from-equilibrium normal-mode perturbations (NMS), providing a controlled arena to assess how curvature supervision impacts both interpolation and extrapolation in reactive MLIPs [33]. In our previous work, full-Hessian-trained MLIPs were shown to improve downstream reactive-chemistry workflows, including molecular dynamics, nudged elastic band calculations, and vibrational-frequency predictions. These results demonstrate that improvements in Hessian accuracy can translate into practical gains beyond aggregate energy, force, and Hessian RMSE [25].

Despite this promise, incorporating full Hessian information presents substantial practical challenges. First, acquiring Hessian data from quantum chemistry calculations can be prohibitively expensive for many electronic structure methods, particularly beyond mean-field levels where analytic second derivatives are substantially more complex or unavailable [21, 22]. Second, training MLIPs with explicit Hessian supervision is significantly more expensive than energy-force training alone [25]. Although gradients can be computed efficiently using reverse-mode automatic differentiation, there is no general method to form all Hessian elements with comparable efficiency [19]. Moreover, storing the Hessian requires  $(3N)^2$  floating-point values per structure, leading to quadratic memory scaling with system size, which rapidly becomes a bottleneck for both data generation and GPU-based neural network training. These combined computational and memory costs severely limit the direct use of full Hessians in large-scale MLIP workflows, motivating alternative strategies that retain essential curvature information without explicitly constructing the Hessian.

In this work, we introduce *Projected Hessian Learning* (PHL) as a stochastic strategy to efficiently incorporate curvature information into the training of s (figure 1). The central idea in PHL is to avoid explicit construction of the full Hessian matrix and instead supervise *projected* curvature information

through Hessian-vector products (HVPs) of the learned PES, as schematized by the randomized probing strategy in figure 1. The complete PHL training loop is summarized in [appendix](#). Within an automatic differentiation framework used to train MLIPs, HVPs can be efficiently computed using forward-over-reverse or reverse-over-reverse differentiation without explicitly forming the Hessian [19, 34]. HVP-based methods are also foundational in large-scale second-order optimization (e.g. Hessian-free and Newton conjugate gradient methods) precisely because they provide curvature access without forming Hessians [35, 36]. In terms of floating-point operations, a single HVP can often be evaluated at a cost comparable to a small constant number of gradient evaluations, regardless of the size of the system  $N$ . [34, 36] Finally, PHL's use of low-dimensional probing subspaces is closely related to randomized 'sketching' and low-rank approximation ideas for implicitly accessed matrices [37, 38].

## 2. Mathematical background

### 2.1. Stochastic estimation of the Hessian loss

We can use the Hutchinson trace estimator to work with HVPs instead of the expensive calculation of the full Hessian matrix [39]. This estimator says that given any matrix  $A$ , one can form an unbiased approximation to the trace using only the matrix-vector product;

$$\text{tr} A \approx v^T A v. \quad (2)$$

Here,  $v$  is a random vector sampled from a distribution with components that are not correlated and have unit variance;

$$\langle v_i v_j \rangle = \delta_{ij}. \quad (3)$$

The unbiased nature of this approximation can be verified by applying the definition of the Kronecker symbol  $\delta_{ij}$  and the linearity of the expectation values  $\langle \cdot \rangle$ ,

$$\text{tr} A = \sum_i A_{ii} = \sum_{ij} \delta_{ij} A_{ij} \approx \sum_{ij} \langle v_i v_j \rangle A_{ij} = \langle v^T A v \rangle. \quad (4)$$

Various random distributions are possible. The essential condition is that the probe vectors have identity covariance,  $\langle v_i v_j \rangle = \delta_{ij}$ , which guarantees an unbiased stochastic trace estimator. Hutchinson's original proposal independently samples each component as  $v_i = \{\pm 1\}$  (the Rademacher distribution). This gives an exact trace estimate when  $A$  is diagonal and, for symmetric  $A$ , has variance no greater than the Gaussian estimator. Another common choice, adopted for the original PHL experiments in this work, is  $v_i \sim \mathcal{N}(0, 1)$ . Gaussian probes also satisfy equation (3) and are rotationally invariant, making them a natural general-purpose choice. The PHL objective is therefore valid with either distribution; their different finite-sample variances are derived and tested in the Supplementary Information.

Equation (2) can be incorporated into MLIP training as follows: First, consider a training loss function that involves energy, force, and Hessian loss terms,

$$\mathcal{L} = \lambda_E \mathcal{L}_E + \lambda_F \mathcal{L}_F + \lambda_H \mathcal{L}_H. \quad (5)$$

The hyperparameters  $\lambda_{\{E,F,H\}}$  describe the relative weighting of these loss terms. Frequently, the  $l_2$  loss on components would be used. For example, a molecule of  $N$  atoms may appear in the loss as

$$\mathcal{L}_H = \frac{1}{(3N)^2} \sum_{i,j=1}^{3N} (\tilde{H}_{ij} - H_{ij})^2. \quad (6)$$

The symbol  $\tilde{H}_{ij}$  will denote the Hessian of the ML-predicted energy  $\tilde{E}$ , which has an implicit dependence on the model parameters  $\theta$ . The symbol  $H_{ij}$  will denote true (reference) quantum chemistry Hessian data. It is our goal to train the model parameters to minimize the loss. In particular, stochastic gradient descent (SGD) requires one to calculate  $\nabla_{\theta} \mathcal{L}_H$ . Clearly, this seems to require performing the complete sum on the components of the  $(3N)^2$  matrix  $\tilde{H}_{i,j}$ .

Let us now review the construction of an efficient stochastic approximation. Equation (6) can be interpreted as a matrix trace,

$$\mathcal{L}_H = \text{tr} A. \quad (7)$$

using the following definitions:

$$A = B^T B \quad (8)$$

$$B = \frac{\tilde{H} - H}{3N}. \quad (9)$$

The matrix  $B$  denotes the scaled error in the Hessian predicted by the ML model  $\tilde{H}$ , while  $B^T B$  forms the corresponding squared-error matrix. One could also write  $A = B^2$  since Hessian matrices are symmetric (i.e. derivatives with respect to atom positions commute). From the matrix trace, one can readily obtain a stochastic approximation,

$$\mathcal{L}_H \approx \hat{\mathcal{L}}_H = v^T A v = |Bv|^2. \quad (10)$$

Back-substitution yields an explicit approximation formula,

$$\mathcal{L}_H \approx \frac{1}{(3N)^2} |\tilde{H}v - Hv|^2, \quad (11)$$

where  $v$  is a random vector of dimension  $3N$  that satisfies equation (3). Importantly, the right-hand side involves only HVPs [19, 34]. In particular, individual elements of the ML-predicted Hessian  $\tilde{H}$  never need to be calculated explicitly. Instead, one can employ, e.g. the `torch.hvp` function to efficiently evaluate  $\tilde{H}v$  as a vector.

### 2.1.1. Randomizing over mini-batches

Generally, a stochastic trace estimator such as equation (11) will be deployed as an average over a large number  $N_v$  of random vectors  $v$ . Since each random vector is associated with an independent and unbiased estimate, the total stochastic error in estimating  $\mathcal{L}_H$  decays like the inverse square root,  $N_v^{-1/2}$  [40, 41].

In practice, training of MLIPs uses SGD over a very large number of mini-batches. It is natural that each mini-batch could employ a new random vector  $v$ . Then, for purposes of SGD training, the overall quality of the stochastic approximation can benefit strongly from having many mini-batches.

### 2.1.2. Random vector distributions

As discussed above, Hutchinson's original proposal was to sample each vector component independently under a uniform distribution,

$$v_i^{\text{Hutch}} = \{\pm 1\}. \quad (12)$$

Recently, in the context of Hessian training, an alternative stochastic estimator was suggested: Work with a randomly selected column of  $(\tilde{H} - H)$  at a time [27, 29, 32]. This method can be cast into our notation taking  $v$  to have a single non-zero component ('hot') at a random index  $c = \{1, \dots, 3N\}$ . This one-hot distribution can be expressed concisely as follows.

$$v_i^{\text{1Hot}} = \sqrt{3N} \delta_{i,c}. \quad (13)$$

To verify the correctness of the 1-hot stochastic estimator, we must check equation (3). Since  $v_i^{\text{1Hot}}$  has a single nonzero element, it is guaranteed

$$\langle v_i^{\text{1Hot}} v_j^{\text{1Hot}} \rangle = 0 \quad (i \neq j). \quad (14)$$

Furthermore, the prefactor  $\sqrt{3N}$  is precisely what is needed to satisfy

$$\begin{aligned} \langle v_i^{\text{1Hot}} v_i^{\text{1Hot}} \rangle &= \frac{1}{3N} \sum_{c=1}^{3N} [v_i^{\text{1Hot}}]^2 \\ &= \frac{1}{3N} \sum_{c=1}^{3N} [\sqrt{3N} \delta_{i,c}]^2 \\ &= \sum_{c=1}^{3N} \delta_{i,c} \\ &= 1. \end{aligned}$$

These two equations demonstrate consistency with equation (3). Therefore, Rademacher, Gaussian, and one-hot vectors all yield valid and unbiased stochastic estimators. The stochastic-error analysis and numerical size-scaling test in the Supplementary Information show that dense Gaussian and Rademacher probes self-average with increasing system size, whereas a single one-hot probe does not show a comparable reduction in relative error along the tested alkane series.

We distinguish between dense trace probes, such as Rademacher or Gaussian vectors, and sparse one-hot probes. Dense probes form linear combinations of many Hessian columns in a single HVP, whereas one-hot probes select a single column. PHL denotes dense identity-covariance probing and is therefore distribution-general; unless otherwise stated, the original PHL results shown in the primary comparisons use Gaussian probes.

### 3. Methodology

We investigate how second-derivative information can be efficiently incorporated into MLIP training. Starting from a chemically diverse quantum-chemical dataset, we train neural-network potentials under multiple supervision strategies that progressively include higher-order derivatives of the PES. Specifically, we compare standard energy-force (E-F) training with approaches that incorporate curvature information either through full Hessians or through HVPs. By keeping the model architecture and the underlying data fixed while varying only the form of derivative supervision, we isolate the impact of curvature information on training dynamics, predictive accuracy, and computational cost.

#### 3.1. Dataset preparation

To evaluate the role of Hessian information in training MLIPs, we employed the same dataset introduced in our previous work [25, 33], which spans both equilibrium and non-equilibrium regions of a PES. All reference calculations were performed with Gaussian16 at the  $\omega$ B97XD/6-31G(d) level of theory [42]. Analytical Hessians were obtained through frequency calculations. The choice of  $\omega$ B97XD provides a reliable treatment of non-covalent interactions and barrier heights, while the 6-31G(d) basis set balances accuracy and computational efficiency and ensures compatibility with the ANI-1x models [3–6, 43, 44].

The dataset is divided into three complementary components:

1. **Benchmark test set:** a collection of 35 087 optimized stationary-point geometries from 11 961 reactions, including RTP states. These structures were excluded from training and serve as a reference for interpolation accuracy.
2. **IRC dataset:** a set of 34 248 geometries sampled from 600 IRC trajectories, providing continuous reaction pathways that probe the curvature of the PES near transition states.
3. **Normal mode sampling (NMS) dataset:** a collection of 62 527 non-equilibrium geometries generated by perturbing intermediate IRC structures along vibrational normal modes for 574 reactions. This dataset is designed to rigorously test the robustness of the extrapolation under large structural distortions.

The molecular systems in these datasets are *small-molecule* geometries, with a typical atom count  $N$  in the *tens* (e.g. median  $N \approx 14$  across the datasets) rather than the hundreds or thousands characteristic of condensed-phase simulations. This point is important because both the stochastic-error analysis and the new alkane-series experiment indicate that dense probing becomes increasingly favorable relative to one-hot probing as  $N$  grows. The latter experiment extends the estimator test to 101 atoms, but contains only one alkane at each size; it supports the proposed scaling mechanism without replacing validation on chemically diverse condensed-phase systems.

For our purposes, these datasets provide diverse coverage of the PES, with the benchmark test set and IRC dataset assessing interpolation performance in chemically relevant regions, and the NMS dataset challenging the models under far-from-equilibrium conditions. These datasets form the common foundation for all the training strategies described below, enabling direct comparison of how different forms of derivative supervision influence learning behavior and generalization.

#### 3.2. Training objective and strategies

Using these datasets, our objective was to quantify the trade-off between predictive accuracy and computational efficiency as progressively higher-order derivative information is introduced into MLIP training. To this end, we evaluated four training schemes that progressively incorporate higher-order derivatives of

the PES into the objective function. The schemes are designed to explore the tradeoffs between accuracy and computational cost; a detailed description of the optimization setup (dataset split, hyperparameters, batching, probe sampling, and other information) is provided in the Supplementary Information under *Training Procedure*. The training schemes are the following:

1. **E-F**: training on total energies  $E$  and atomic forces  $F$ . This baseline reflects standard practice in the development of MLIPs.
2. **E-F-HVP (one-column)**: training on energies, forces, and Hessian–vector products (HVP), where  $Hv = \nabla^2 E(\mathbf{R})v$  is evaluated with a single one-hot probe vector  $v^{\text{1Hot}}$  aligned with a coordinate axis. This approach introduces curvature information at minimal additional cost and has been recently used for training MLIPs to Hessian information [29, 32].
3. **E-F-HVP (PHL)**: training on energies, forces, and HVPs using PHL, where  $Hv = \nabla^2 E(\mathbf{R})v$  is evaluated using dense identity-covariance probing vectors. The original experiments use Gaussian probes,  $v_i \sim \mathcal{N}(0, 1)$ . Follow-up experiments repeat the same training with Rademacher probes,  $v_i \in \{-1, +1\}$ , using matched seeds.
4. **E-F-H**: training on energies, forces, and full Hessians. This provides the most complete curvature information but is associated with a prohibitive increase in computational cost.

In summary, E-F-H provides the most complete curvature-supervision benchmark but is computationally expensive, while E-F serves as the lowest-cost baseline. The two HVP-based methods offer intermediate strategies that capture much of the benefit of Hessian training while remaining computationally efficient.

In both HVP-based schemes, we considered two vector protocols: (i) *randomized-vector* training, in which new probe vectors are resampled at every minibatch, and (ii) *fixed-vector* training, in which a single probe vector per configuration is retained throughout training. The fixed-vector setting serves as a controlled surrogate for scenarios in which the ground-truth Hessian is not known in full, but itself sampled.

Finally, we examine both the running time of the machine-learning model training and the cost of generating the underlying quantum mechanical reference data to assess the efficiency and benefits of the different training schemes. We compared the cost of computing energies, forces, full Hessians and HVPs using DFT. To allow fair comparison between training schemes, all models were optimized using a unified loss framework that differs only in how curvature information is incorporated.

### 3.3. Loss function formulation

The total training loss was defined as a weighted sum of errors in predicted energies, forces, and, when included, Hessian information. Two formulations were used depending on whether the model was trained with full Hessians or with HVPs:

$$\mathcal{L}_{\text{full}} = \mathcal{L}_E(E^{\text{pred}}, E^{\text{ref}}) + \lambda_F \mathcal{L}_F(F^{\text{pred}}, F^{\text{ref}}) + \lambda_H \mathcal{L}_H(H^{\text{pred}}, H^{\text{ref}}), \quad (15)$$

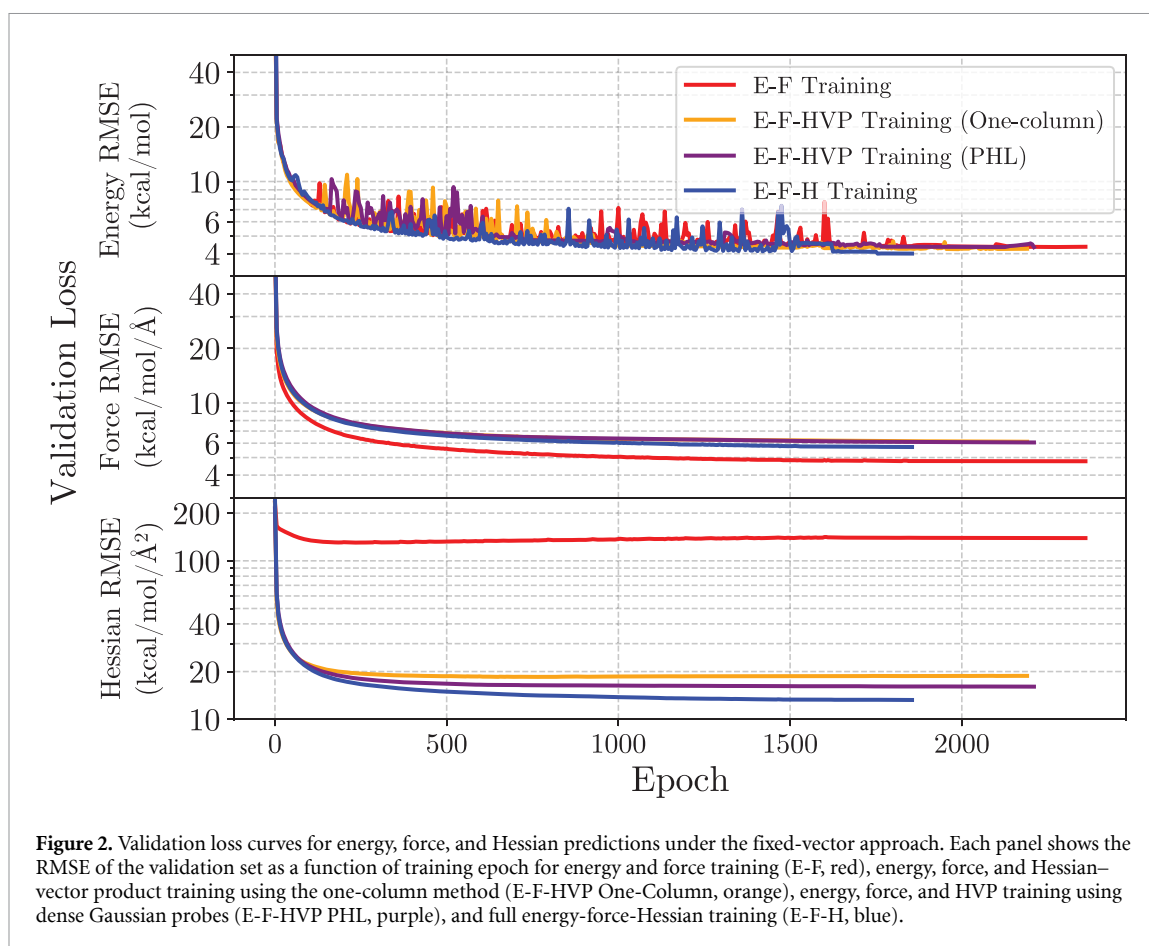
$$\mathcal{L}_{\text{HVP}} = \mathcal{L}_E(E^{\text{pred}}, E^{\text{ref}}) + \lambda_F \mathcal{L}_F(F^{\text{pred}}, F^{\text{ref}}) + \lambda_H \hat{\mathcal{L}}_H(H^{\text{pred}}v, H^{\text{ref}}v). \quad (16)$$

Here,  $E$  represents the total molecular energies,  $F$  represents atomic forces,  $H$  represents the Hessian matrices, and  $v$  represents probe vectors. In the full-Hessian formulation, the model is trained against all second derivatives, whereas in the HVP formulation, only products of the Hessian with probe vectors are included. The weights ( $\lambda_F = 0.30$ ) and ( $\lambda_H = 0.09$ ) were tuned to balance the relative contributions of forces and Hessian information against energies. Typically, energies set the absolute scale, while forces and Hessian terms are given smaller weights to ensure they contribute comparably during optimization. All models were trained under identical optimization settings, with differences arising solely from the form of derivative supervision.

## 4. Results and discussion

### 4.1. Validation loss convergence

The convergence behavior of the different training schemes was examined using validation RMSE curves for energies, forces, and Hessians under the fixed-vector approach (figure 2). These curves provide insight into how curvature information influences optimization dynamics, training stability, and the relative behavior of one-column and dense Gaussian HVP estimators (PHL). The validation results for the



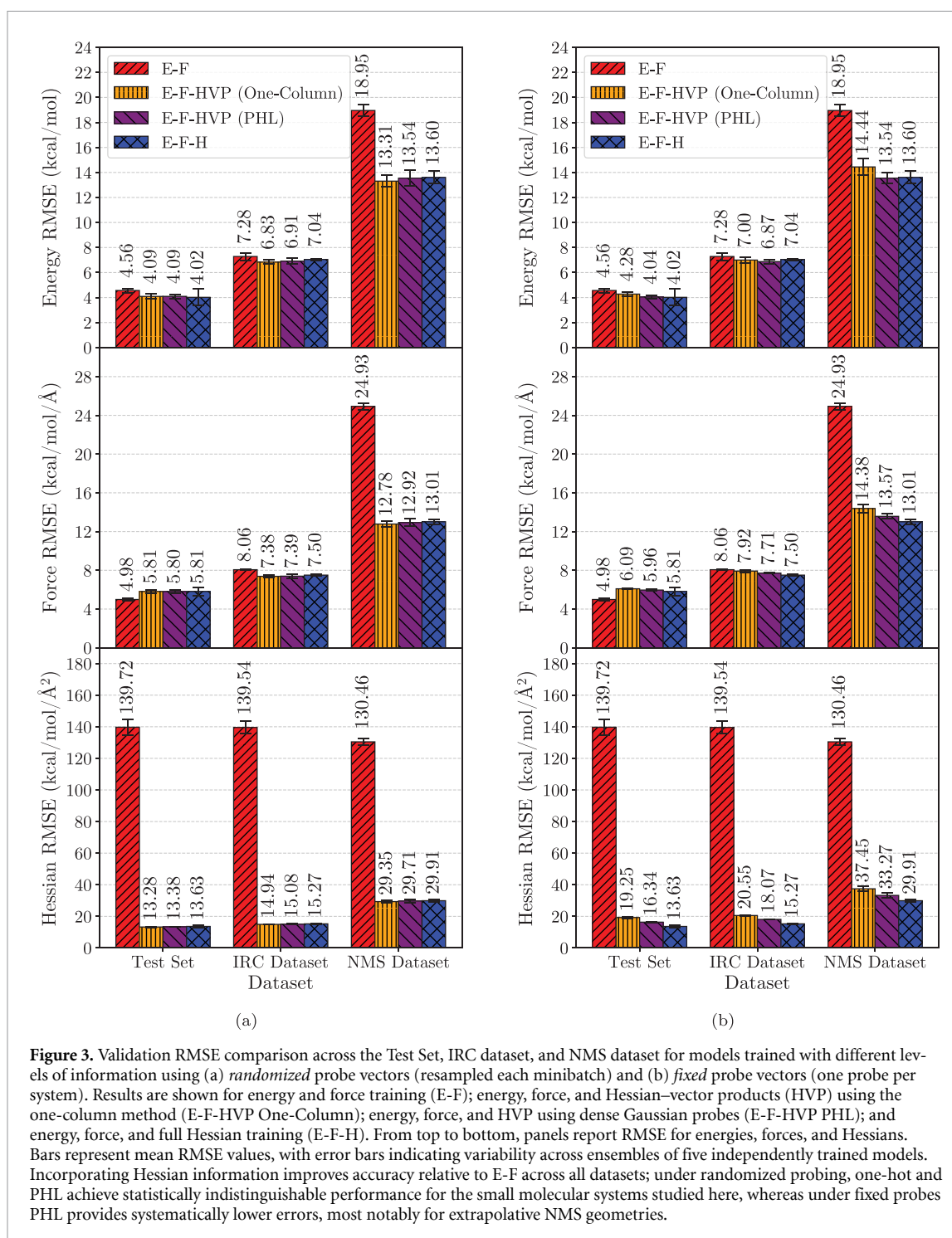
randomized-vector experiments reached comparable final RMSE values for the characteristic system size of our training dataset (a median of 14 atoms per system) and are therefore omitted from this section for brevity; their performance in generalization is analyzed separately below.

For energy errors, all training schemes eventually converge to similar RMSE validation levels. However, clear differences appear during the early and intermediate stages of training. Methods incorporating curvature information exhibit reduced fluctuations in validation loss after approximately 700 epochs, resulting in smoother and more stable convergence trajectories. Full Hessian training (E-F-H) shows the most stable behavior early in training and converges more rapidly, while both HVP-based approaches achieve greater stability than the E-F baseline during the intermediate and late stages of training.

This smoother convergence suggests that the incorporation of second-derivative information regularizes the optimization process, reducing the sensitivity to stochastic noise and sharp variations in the loss landscape. Such stabilization effects have been widely observed in both machine learning and atomistic modeling [45, 46], where smoother optimization trajectories are often associated with improved generalization and robustness, even when the final training errors are similar.

For force errors, E-F training achieves the lowest absolute validation RMSE, consistent with its objective function placing the greatest emphasis on force accuracy. Importantly, introducing second-derivative information, either via full Hessians or HVP estimators, does not degrade force prediction accuracy. Both E-F-HVP models closely track the convergence behavior of E-F-H, indicating that curvature information can be incorporated without compromising force fidelity. In particular, although E-F achieves slightly lower force RMSE in the validation set, this does not necessarily translate to improved generalization, particularly for configurations outside of the training distribution, as discussed in the following section.

For Hessian errors, models trained only on energies and forces exhibit large and poorly converged validation errors, indicating that curvature information is not reliably recovered from first-order data alone. Both HVP-based approaches recover much of the smoother convergence and lower Hessian error obtained with full-Hessian supervision. The quantitative differences among datasets and between one-column and dense probing are examined in the following section.



Crucially, HVP-based methods recover much of the curvature accuracy and training stability of full Hessian supervision at a dramatically reduced computational cost, making them effective in regimes where explicit Hessian evaluation is cost prohibitive. Overall, HVP-based training captures most of the optimization and generalization benefits of full Hessian inclusion, with dense PHL offering improved robustness and accuracy when only a single fixed HVP per molecular system is available.

#### 4.2. Predictive accuracy across datasets

To evaluate generalization performance, we compared the final RMSE values of all models on the Test Set, the IRC dataset, and the NMS dataset. Figure 3 summarizes the results under the randomized-vector and fixed-vector approaches with separate panels for the energy, force, and Hessian predictions.

#### 4.2.1. Randomized HVP Vectors per minibatch

When the HVP probe vector is randomized in each minibatch step (figure 3(a)), the performance of the one-column and PHL estimators is nearly indistinguishable across all datasets and all training targets for the system sizes in the training data.

- **Energy RMSE:** under randomized-vector training, both HVP estimators substantially improve upon the E-F baseline and approach full Hessian accuracy. On the NMS dataset, the energy RMSE is reduced by approximately 29% compared to E-F for both one-column and PHL estimators, with only a difference of 1%–2% between them. Similar trends are observed on the Test and IRC datasets, indicating that stochastic curvature sampling provides sufficient coverage for accurate energy prediction.
- **Force RMSE:** although E-F remains marginally optimal for in-domain force prediction, HVP-based methods closely match full Hessian performance and substantially outperform E-F on extrapolative NMS geometries. Specifically, the force RMSE on NMS is reduced by approximately 48%–49% relative to E-F for both HVP estimators, with a separation of less than 2% between the one-column and PHL approaches.
- **Hessian RMSE:** HVP training dramatically improves Hessian prediction accuracy compared to E-F, reducing the NMS Hessian RMSE by approximately 77% for both estimators. The one-column and PHL methods exhibit nearly identical performance in this randomized setting, differing only by  $\sim 1\%$ , confirming that minibatch-level randomization provides sufficiently rich curvature sampling to approximate full Hessian supervision.

Randomized HVP vectors therefore enable broad curvature coverage during training, yielding performance nearly indistinguishable from full Hessian training for energies, forces, and Hessians, while avoiding the substantial computational cost of explicit Hessian supervision. The size-scaling experiment below further shows that dense probes become more precise estimators as  $N$  grows.

#### 4.2.2. Fixed HVP vectors per system

The fixed-vector setting (figure 3(b)) provides a more realistic scenario in which only one HVP per system is available from quantum chemistry. Under these conditions, the differences between the one-column and PHL estimators become more pronounced.

- **Energy RMSE:** All Hessian-informed methods improve energy RMSE relative to E-F training across all datasets, with gains that are largest on extrapolative geometries. On the Test set, energy RMSE decreases by 6.1% with the one-column estimator and 10.3% with PHL, compared to E-F (11.8% for full Hessian training). On the IRC dataset, the reductions are smaller but consistent: 3.8% for one-column and 5.6% for PHL. On the NMS dataset, the improvements are substantial: one-column reduces the energy RMSE by 23.8% and PHL by 28.5% relative to E-F, with PHL achieving a 6.2% reduction in RMSE compared to one-column (and essentially matching complete Hessian training, 28.2%).
- **Force RMSE:** The benefit of curvature supervision is most apparent for intermediate or NMS configurations. Although E-F training attains the lowest RMSE on the Test set, both HVP methods match force accuracy on the IRC geometries and substantially improve force accuracy on NMS geometries. Compared to E-F, the one-column and PHL estimators reduce the force RMSE of the NMS dataset by 42.3% and 45.6%, respectively (47.8% for full Hessian training). The PHL estimator consistently achieves a lower RMSE than the one-column estimator across datasets, with a 2.1% reduction in RMSE on the Test set, 2.7% on the IRC set, and 5.6% on the NMS set.
- **Hessian RMSE:** The strongest separation between estimators appears for Hessian prediction. Relative to E-F training, Hessian RMSE is reduced by 86.2% with one-column estimators and 88.3% with PHL estimators on the Test set; 85.3% and 87.0% on IRC; and 71.3% and 74.5% on NMS. The PHL estimator consistently outperforms the one-column estimator across all datasets, with Hessian RMSE lower by 15.1% on the Test set, 12.1% on IRC, and 11.2% on NMS.

When only a single HVP per molecular configuration is available, the PHL estimator yields errors that are systematically lower than those of the one-column approach across all predicted properties. On the extrapolative NMS dataset, PHL reduces the energy RMSE by 6.2%, the force RMSE by 5.6%, and the Hessian RMSE by 11.2% relative to the one-column estimator. These improvements indicate that dense

**Table 1.** Paired *t*-test results comparing RMSE values from the one-column and PHL estimators under the randomized-vector training approach. The mean difference is defined as (one-column—PHL). Negative values indicate lower RMSE for the one-column method. Mean differences are reported in kcal mol<sup>−1</sup> for energy, kcal mol<sup>−1</sup> Å<sup>−1</sup> for forces, and kcal mol<sup>−1</sup> Å<sup>−2</sup> for Hessians. Asterisks denote nominal significance ( $p < 0.05$ ; unadjusted for multiple comparisons).

| Property | Dataset | Mean Difference | <i>p</i> -value | Significance |
|----------|---------|-----------------|-----------------|--------------|
| Energy   | Test    | −0.0039         | 0.950           | —            |
| Energy   | IRC     | −0.0775         | 0.348           | —            |
| Energy   | NMS     | −0.2320         | 0.342           | —            |
| Force    | Test    | 0.0134          | 0.832           | —            |
| Force    | IRC     | −0.0198         | 0.732           | —            |
| Force    | NMS     | −0.1444         | 0.195           | —            |
| Hessian  | Test    | −0.1051         | 0.405           | —            |
| Hessian  | IRC     | −0.1468         | 0.166           | —            |
| Hessian  | NMS     | −0.3608         | 0.189           | —            |

random vectors provide broader coordinate coverage under fixed-vector constraints. While the one-column estimator remains a computationally inexpensive alternative, its reliance on a single coordinate direction limits performance in data-sparse regimes and large system sizes.

Although the present study evaluates downstream impact primarily through energy, force, and Hessian errors, these errors are directly connected to properties controlled by local PES curvature, including vibrational frequencies, transition-state modes, and reaction-path stability. In prior work using the same Hessian-informed MLIP framework, full-Hessian supervision was shown to improve molecular dynamics, nudged elastic band calculations, and vibrational-frequency predictions [25]. The present PHL results therefore address the complementary methodological question of how much of this full-Hessian curvature information can be recovered at HVP-level cost; the fixed-vector results indicate that dense projected probes provide a lower-cost route toward the same class of curvature-sensitive applications.

#### 4.3. Statistical significance of RMSE differences

To quantify whether the one-column and PHL estimators produce statistically distinguishable predictive accuracy, we conducted paired *t*-tests on the RMSE values from five matched training seeds for each method. The tests were performed separately for the randomized-vector and fixed-vector regimes, allowing us to isolate the effect of the vector-sampling strategy on estimator behavior. The results are summarized in tables 1 and 2, where the mean RMSE difference is defined as (one-column RMSE—PHL RMSE). Thus, negative values indicate lower RMSE for the one-column estimator, while positive values indicate lower RMSE for the PHL estimator. Reported *p*-values are unadjusted for multiple comparisons, so  $p < 0.05$  denotes nominal significance.

##### 4.3.1. Randomized HVP vectors per minibatch

Table 1 reports the results of the paired *t*-test for experiments in which the HVP vector was randomly resampled in each minibatch. Under this configuration, no statistically significant differences were observed between the two estimators across any of the datasets or properties used (all  $p > 0.05$ ).

Across energies, forces, and Hessians, the mean RMSE differences are small and fluctuate around zero, consistent with the interpretation that randomized stochastic sampling effectively averages out directional bias for our characteristic system sizes. Randomized vectors ensure broad coverage of curvature directions throughout training, allowing both estimators to recover similarly accurate approximations to the Hessian.

These results indicate that when HVP vectors vary throughout training, the one-column and PHL estimators perform equivalently, and neither method shows a statistically significant advantage for these system sizes.

##### 4.3.2. Fixed HVP Vectors per System

The corresponding results of the paired *t*-test for fixed-vector experiments are shown in table 2. In the fixed-vector regime, a single HVP vector is assigned to each molecule and reused across all epochs. This setting simulates scenarios in which only one HVP is available per system from quantum chemical calculations. Under this more restrictive condition, nominally significant differences emerge between the two estimators for several datasets and training targets.

**Table 2.** Paired *t*-test results comparing RMSE values from the one-column and PHL estimators under the fixed-vector training approach. The mean difference is defined as (one-column—PHL). Negative values indicate lower RMSE for the one-column method. Mean differences are reported in kcal mol<sup>−1</sup> for energy, kcal mol<sup>−1</sup> Å<sup>−1</sup> for forces, and kcal mol<sup>−1</sup> Å<sup>−2</sup> for Hessians. Asterisks denote nominal significance ( $p < 0.05$ ; unadjusted for multiple comparisons).

| Property | Dataset | Mean Difference | <i>p</i> -value | Significance |
|----------|---------|-----------------|-----------------|--------------|
| Energy   | Test    | 0.2370          | 0.063           | —            |
| Energy   | IRC     | 0.1303          | 0.399           | —            |
| Energy   | NMS     | 0.8989          | 0.006           | *            |
| Force    | Test    | 0.1268          | 0.097           | —            |
| Force    | IRC     | 0.2076          | 0.013           | *            |
| Force    | NMS     | 0.8086          | 0.006           | *            |
| Hessian  | Test    | 2.9105          | 0.0000 029      | *            |
| Hessian  | IRC     | 2.4722          | 0.00 022        | *            |
| Hessian  | NMS     | 4.1780          | 0.005           | *            |

- **Energy RMSE:** For the Test and IRC datasets, the mean differences are not statistically significant ( $p > 0.05$ ), indicating comparable energy predictions. For the NMS dataset, PHL yields lower energy RMSE ( $p = 0.006$  unadjusted), a nominally significant difference consistent with better extrapolative capability.
- **Force RMSE:** PHL yields lower force RMSE on the IRC dataset ( $p = 0.013$  unadjusted) and the NMS dataset ( $p = 0.006$  unadjusted), both nominally significant differences. This suggests that its broader curvature sampling improves first-derivative accuracy in intermediate and far-from-equilibrium geometries, where directional bias from a single one-column vector becomes limiting.
- **Hessian RMSE:** Across all datasets, PHL yields lower Hessian RMSE ( $p < 0.01$  unadjusted for all comparisons), with each difference nominally significant. These results support the conclusion that dense Gaussian vectors provide better curvature information than one-column probes when only one vector per molecule is available.

This analysis indicates that when curvature information is limited to a single HVP per system, dense Gaussian PHL consistently outperforms the one-column method, especially for the NMS dataset and for Hessian predictions. This result is attributed to simultaneous coverage of all Cartesian coordinates and an isotropic second moment, rather than to the specific Gaussian distribution alone.

#### 4.4. Gaussian versus Rademacher PHL probes

The original PHL models used Gaussian probes because they satisfy the identity-covariance condition required by equation (3), are rotationally invariant, and therefore provide an unbiased estimate of the same full-Hessian loss. To determine whether this choice affected the reported accuracy, we repeated both the randomized- and fixed-probe PHL experiments with Rademacher vectors using the same five training seeds. All models were evaluated with complete reference Hessians on the Test, IRC, and NMS datasets; thus, the comparison is not based on a probe-specific validation loss. Supplementary figure S1 shows the most challenging NMS results, together with the complete paired analysis.

None of the 18 paired comparisons (three properties, three datasets, and two probe protocols) shows a statistically significant Gaussian–Rademacher difference at the 0.05 level, and every 95% confidence interval includes zero. For randomized probes on NMS, the Rademacher means are lower by 1.1%, 2.1%, and 3.1% for energy, force, and Hessian RMSE, respectively, but the corresponding paired-test *p*-values are 0.671, 0.197, and 0.263. With fixed probes, neither distribution is consistently better ( $p = 0.920$ , 0.983, and 0.855 on the same metrics). With five seeds, these null results do not establish exact equivalence or rule out a small effect; they show that probe distribution did not measurably change trained-model accuracy at the resolution of this experiment.

This finding both supports the validity of the original Gaussian calculations and motivates a refined practical recommendation. Gaussian and Rademacher probes optimize the same expected PHL objective, explaining their similar final accuracy after many stochastic updates. Rademacher probes nevertheless have lower or equal single-probe trace-estimator variance for symmetric matrices and fixed component

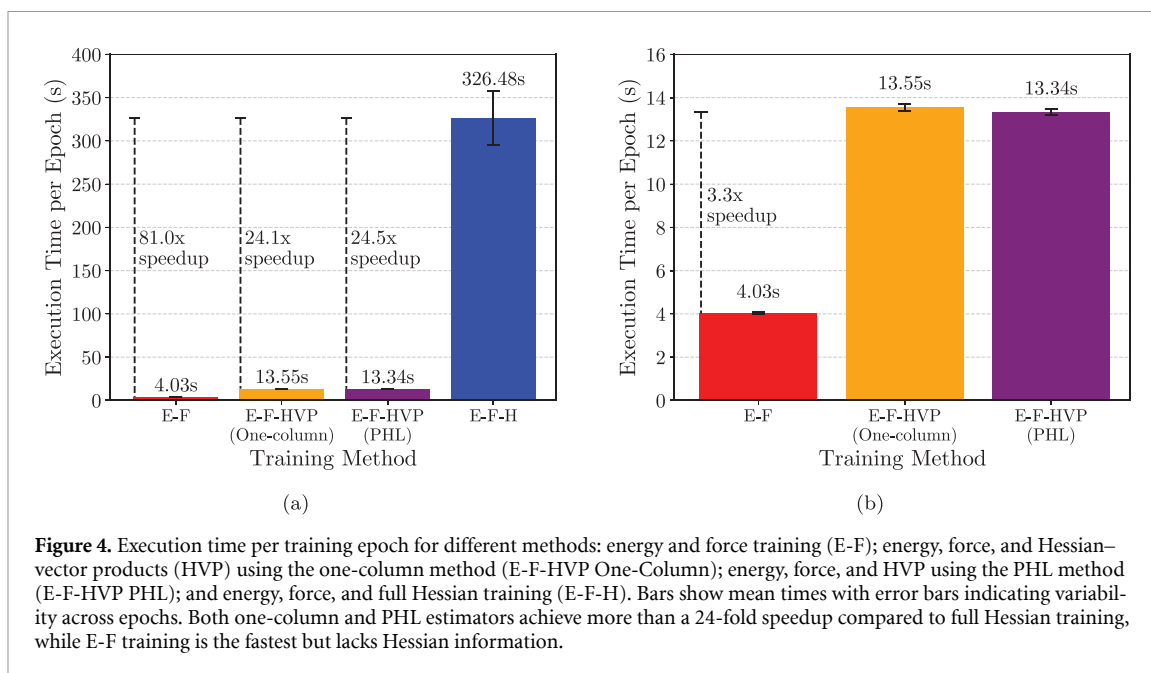


**Figure 4.** Execution time per training epoch for different methods: energy and force training (E-F); energy, force, and Hessian–vector products (HVP) using the one-column method (E-F-HVP One-Column); energy, force, and HVP using the PHL method (E-F-HVP PHL); and energy, force, and full Hessian training (E-F-H). Bars show mean times with error bars indicating variability across epochs. Both one-column and PHL estimators achieve more than a 24-fold speedup compared to full Hessian training, while E-F training is the fastest but lacks Hessian information.

magnitudes. The latter property bounds every Cartesian displacement in finite-difference HVP generation. We therefore retain the Gaussian results as unbiased and empirically robust, while recommending Rademacher probes as the default for future PHL implementations when only a small number of probes or finite-displacement references are available.

#### 4.5. Computational efficiency

A key motivation for incorporating HVPs into MLIP training is the ability to reap the benefits of second-derivative information without incurring the cost of full Hessian evaluations. To quantify this advantage, we assessed computational efficiency at two levels: (1) MLIP training runtime per epoch in figure 4 and (2) quantum-chemistry cost scaling for computing energies, forces, Hessians, and HVPs in figure 6.

##### 4.5.1. MLIP training runtime

Figure 4 reports the execution time per epoch for all training schemes. Full Hessian training (E-F-H) is by far the most computationally demanding, requiring over an order of magnitude more time per epoch than the other methods. This cost arises from the need to backpropagate through the complete Hessian matrix, which involves repeated second-derivative evaluations.

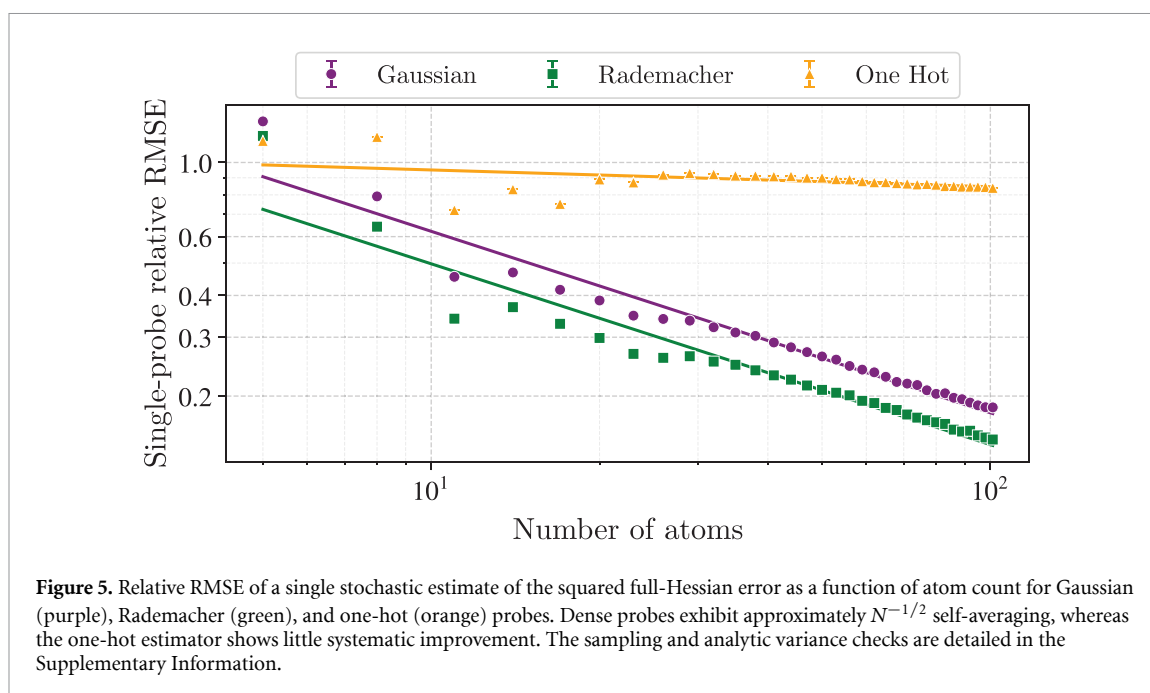
In contrast, both HVP-based approaches (E-F-HVP One-Column and E-F-HVP PHL) achieve epoch times of approximately 13 s, representing a 24-fold speedup relative to full Hessian training. Importantly, both estimators preserve most of the predictive gains afforded by full Hessians, providing a near-optimal tradeoff between speed and accuracy.

The nearly identical epoch times for one-column and PHL training are expected, because both methods compute one HVP per probe and differ only in the choice of the vector  $\mathbf{v}$ . The runtime comparison is therefore intended to show that HVP supervision, regardless of probe distribution, is much cheaper than full Hessian supervision. The advantage of PHL over one-column probing arises from greater curvature coverage at the same HVP cost.

Energy–force training (E-F) remains the fastest configuration at roughly 4 s per epoch, but lacks any curvature information. The HVP methods therefore occupy a favorable middle ground: only three times slower than E-F training but dramatically more informative. This comparison confirms that incorporating curvature information via HVPs is highly practical in large-scale MLIP training pipelines, particularly when full Hessians would be computationally prohibitive.

##### 4.5.2. Estimator accuracy with system size

The theoretical advantage of dense probing concerns the statistical precision of the stochastic loss, not necessarily the final accuracy of a model trained over many minibatches. We tested this distinction directly using the same full-Hessian-trained model and DFT Hessian-error matrix for Gaussian,



Rademacher, and one-hot estimators along an alkane series from 5 to 101 atoms. At each size, 20 000 draws were used to characterize the error of a *single* probe.

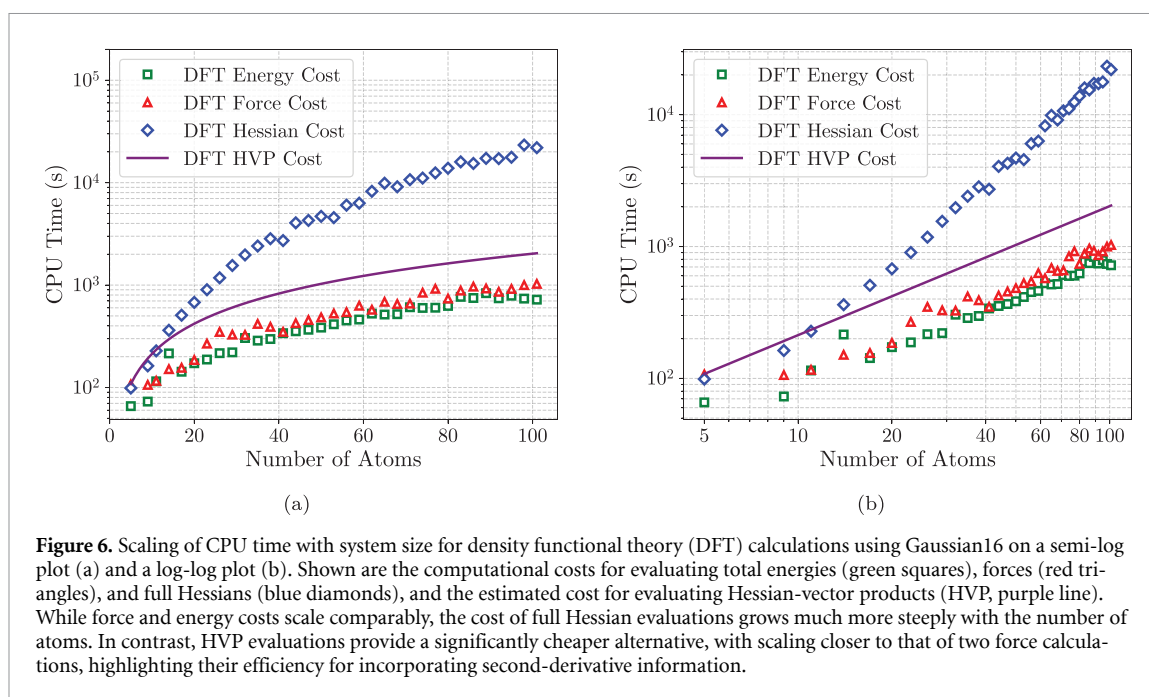
As shown in figure 5, fitted relative-RMSE exponents are  $-0.543$  for Gaussian probes ( $R^2 = 0.950$ ),  $-0.541$  for Rademacher probes ( $R^2 = 0.918$ ), and  $-0.050$  for one-hot probes ( $R^2 = 0.183$ ). At 101 atoms, the single-probe relative RMSE is 0.185, 0.148, and 0.839, respectively. Averaging independent probes reduces these values as  $m^{-1/2}$ ; approximately 14 Gaussian, 9 Rademacher, or 282 one-hot probes would therefore be required to reach 5% relative RMSE for this largest molecule. The close agreement between empirical and analytic variances confirms unbiased sampling and identifies removal of diagonal norm fluctuations as the source of the Rademacher advantage. Because the series contains only one alkane at each atom count, the fitted exponents demonstrate the expected mechanism for this controlled family rather than a universal chemical scaling law.

#### 4.5.3. Quantum-chemistry cost scaling

To contextualize the MLIP results with first-principles cost considerations, figure 6 presents the CPU scaling for evaluating energies, forces, full Hessians, and approximate HVP costs using Gaussian16. Energy and force calculations show similar scaling behavior as system size increases, reflecting their reliance on first derivatives of the electronic energy. However, full Hessian evaluations grow much more steeply. This superlinear scaling rapidly dominates computational budgets for systems beyond a few dozen atoms, limiting the practical generation of high-quality second-derivative datasets.

By contrast, HVP evaluations scale much closer to forces. A single HVP can be computed using two force-like operations via forward-over-backward or backward-over-backward automatic differentiation. Alternatively, when the probing direction  $v$  is fixed, the same quantity can be obtained from finite differences of forces by evaluating  $\mathbf{F}(\mathbf{R} \pm \epsilon v)$  and forming  $Hv \approx -[\mathbf{F}(\mathbf{R} + \epsilon v) - \mathbf{F}(\mathbf{R} - \epsilon v)]/(2\epsilon)$ , which again requires only two force evaluations. As a result, a single HVP is expected to cost on the order of two force evaluations, making it orders of magnitude cheaper than computing a full Hessian matrix. Because HVPs capture directional curvature information at such a low cost, they are an attractive surrogate for full Hessians both at the quantum-chemistry level (for data generation) and at the MLIP training level (for incorporating curvature into the loss function).

We validated this construction against analytic DFT Hessians for 12 structures (four each from RTP, IRC, and NMS; 10–17 atoms), two paired directions per structure, and six directional step sizes. For steps from  $10^{-4}$  to  $10^{-3}$  Å, the median relative  $L_2$  error remains on a plateau near  $1.3\text{--}1.5 \times 10^{-4}$  for both distributions; above  $10^{-3}$  Å, it rises consistently with centered-difference truncation error. At  $\epsilon = 10^{-3}$  Å, the Gaussian and Rademacher median errors are  $1.47 \times 10^{-4}$  and  $1.31 \times 10^{-4}$ , respectively. We therefore identify  $10^{-3}$  Å as the largest safe step in the tested range, rather than claiming a universal optimum. Rademacher probes add the practical benefit that each Cartesian displacement is exactly



bounded by  $\epsilon$ , whereas Gaussian components are unbounded. The complete step sweep and its limitations are given in the supplementary information.

This scaling difference becomes especially important for condensed-phase systems and supercells. A full Cartesian Hessian contains  $(3N)^2 = 9N^2$  entries per configuration, whereas a single HVP contains only  $3N$  entries. Thus, the storage required for full Hessian labels grows quadratically with atom count, while fixed-vector HVP labels grow linearly. For example, a 1000-atom supercell would require storing  $9 \times 10^6$  Hessian elements per configuration, compared with only  $3 \times 10^3$  elements for one projected HVP. Similarly, finite-difference construction of a complete Hessian requires displacing many coordinate directions, whereas a fixed projected HVP requires only the two force evaluations in the expression above. These memory and data-generation considerations motivate projected-curvature supervision for large molecular clusters, periodic cells, surfaces, and defect-containing supercells where explicit Hessian construction is impractical.

## 5. Conclusions

We introduced PHL, a scalable curvature-supervision framework for MLIPs that replaces explicit Hessian construction with fast HVP probes. Using a chemically diverse dataset that spans reactants, products, transition states, IRC, and normal-mode-sampled (NMS) geometries, we compared four training strategies of increasing physical fidelity: energy-force (E-F) training, HVP-based (E-F-HVP) training using one-column or PHL estimators, and full Hessian (E-F-H) training.

Incorporating curvature information produces smoother validation trajectories and improves predictive accuracy. Across Test, IRC, and NMS datasets, the HVP-trained models substantially outperform the E-F baselines, approach full-Hessian training accuracy for the energies and forces, and reduce Hessian errors by approximately 77% on extrapolative NMS geometries. Under randomized-vector training, both HVP-based estimators achieve nearly identical performance for our system sizes, reducing the NMS energy RMSE by  $\sim 29\%$ , the force RMSE by  $\sim 48\%$  and the Hessian RMSE by  $\sim 77\%$  relative to E-F. In contrast, under fixed-vector conditions representative of data-limited regimes, dense Gaussian PHL outperforms one-hot probing, yielding additional reductions of 6.2% in energy RMSE, 5.6% in force RMSE, and 11.2% in Hessian RMSE on NMS geometries, with the corresponding paired comparisons nominally significant at  $p < 0.05$ .

Computationally, full Hessian training is dramatically more expensive, requiring on average 326.5 s per epoch, compared to 13.6 s and 13.3 s for one-column probing and dense PHL, respectively, corresponding to  $\sim 24\times$  speedups relative to E-F-H while incurring only a modest overhead compared to standard E-F training. A centered finite difference of two DFT force calculations reproduces analytic

HVPs with median relative errors near  $10^{-4}$  for directional steps from  $10^{-4}$  to  $10^{-3}$  Å in the tested 10–17 atom structures. Along a separate 5–101 atom alkane series, the relative single-probe estimator error decreases approximately as  $N^{-1/2}$  for both Gaussian and Rademacher probes, whereas one-hot error shows little systematic improvement.

The paired Gaussian–Rademacher retraining experiment clarifies the role of probe distribution. Both distributions are unbiased for the same PHL objective, and none of 18 paired model-accuracy comparisons is significant with five matched seeds. The original Gaussian choice is therefore methodologically valid and did not measurably compromise the reported accuracy. Rademacher probes nevertheless have lower single-probe estimator variance in theory and in the size-scaling experiment, and their fixed component magnitudes bound finite-difference displacements. We consequently recommend Rademacher probes as the practical default for future PHL data generation and training, without interpreting the present paired training results as evidence of superior final model accuracy.

These results establish PHL as an efficient and scalable alternative to full Hessian supervision, delivering substantial accuracy gains over energy-force models at a greatly reduced cost. The PHL perspective also naturally connects to recent PFT work, which uses stochastic curvature supervision to improve force-constant (phonon) accuracy in periodic materials; PHL provides a general projected-curvature framework that can be applied to materials settings where curvature governs vibrational and elastic response, and where large supercells make explicit Hessians impractical.

Future work will test the observed estimator scaling across chemically diverse large systems, develop adaptive multi-probe strategies, and integrate projected-curvature supervision with active learning and uncertainty quantification. The present alkane series and finite-difference sweep are controlled validations, not universal guarantees for condensed phases, soft modes, or other electronic-structure settings; production displacement steps should be validated for the intended method and system class.

## Acknowledgments

This work is supported by the U S Department of Energy, Office of Basic Energy Sciences (FWP: LANLE8AN and FWP: LANLE3F2) and by the U S Department of Energy through Los Alamos National Laboratory. Los Alamos National Laboratory is operated by Triad National Security, LLC, for the National Nuclear Security Administration of the U S Department of Energy Contract No. 892333218NCA000001. This research used resources provided by the LANL CAI-1 Darwin cluster at LANL. This work was supported in part through computational resources and services provided by the Institute for Cyber-Enabled Research at Michigan State University (MSU). This work used the MSU Data Machine, which is supported through the NSF Campus CyberInfrastructure program through Grant #2200792.

## Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: <https://doi.org/10.6084/m9.figshare.29189858> [33].

Supplementary Information available at <https://doi.org/10.1088/2632-2153/acac9b/data1>.

## Code availability

The code used to train and evaluate the ANI models in this paper, including notebooks and instructions for accessing the datasets, is available at: <https://github.com/Austinrg14/PHL>.

In addition, we implemented Hessian and Hessian–vector product (HVP) training functionality in the HIPPYNN framework to enable curvature-supervised training of HIP-NN models (including full-Hessian and projected-curvature/PHL training workflows).

HIPPYNN is available through the Los Alamos National Laboratory (LANL) GitHub repository: <https://github.com/lanl/hippynn>. An example script demonstrating our implementation is provided at `hippynn/examples/hessian_training.py`.

## Appendix. Projected Hessian learning training algorithm

**Algorithm 1.** Projected Hessian Learning (PHL) for second-order derivative training of MLIPs using Hessian-Vector Products (HVPs).

**Input:** Dataset  $\mathcal{D} = \{(\mathbf{R}, E, \mathbf{F}, \mathcal{C})\}$ , where  $\mathbf{R} \in \mathbb{R}^{3N}$ ,  $E$  is reference energy,  $\mathbf{F} = -\nabla_{\mathbf{R}} E$  is reference force. Curvature data  $\mathcal{C}$  may contain one or more HVP pairs  $(v, Hv)$ .

**Input:** MLIP  $\tilde{E}_{\theta}(\mathbf{R})$ , loss weights  $\lambda_E, \lambda_F, \lambda_H$ , probes per structure  $K$ , step size  $\eta$ .

**Output:** Trained parameters  $\theta$ .

Initialize parameters  $\theta$

**While** not converged **do**

    Sample minibatch  $\mathcal{B} \subset \mathcal{D}$

$\mathcal{L} \leftarrow 0$

**foreach**  $(\mathbf{R}, E, \mathbf{F}, \mathcal{C}) \in \mathcal{B}$  **do**

$\tilde{E} \leftarrow \tilde{E}_{\theta}(\mathbf{R})$

$\tilde{\mathbf{F}} \leftarrow -\nabla_{\mathbf{R}} \tilde{E}$

$\mathcal{L}_E \leftarrow \|\tilde{E} - E\|^2$

$\mathcal{L}_F \leftarrow \frac{1}{3N} \|\tilde{\mathbf{F}} - \mathbf{F}\|^2$

        // PHL curvature loss: average over  $K$  Hessian-vector probes

$\hat{\mathcal{L}}_H \leftarrow 0$

**for**  $k \leftarrow 1$  **to**  $K$  **do**

**if**  $(v_k, Hv_k) \in \mathcal{C}$  **then**

$v \leftarrow v_k$

$y \leftarrow Hv_k$

**else**

                // Generate an identity-covariance probe and obtain reference  $y = Hv$

                Sample  $v_i \in \{-1, +1\}$  (recommended) or  $v \sim \mathcal{N}(0, I)$

                Obtain reference  $y \leftarrow Hv$  using the available QC procedure

                // Compute predicted HVP without forming  $\tilde{H}$

$\tilde{y} \leftarrow \tilde{H}v$  via AD HVP (e.g.  $\tilde{y} = -\nabla_{\mathbf{R}}(\tilde{\mathbf{F}})v$ )

$\hat{\mathcal{L}}_H \leftarrow \hat{\mathcal{L}}_H + \frac{1}{(3N)^2} \|\tilde{y} - y\|^2$

$\hat{\mathcal{L}}_H \leftarrow \frac{1}{K} \hat{\mathcal{L}}_H$

$\mathcal{L} \leftarrow \mathcal{L} + \lambda_E \mathcal{L}_E + \lambda_F \mathcal{L}_F + \lambda_H \hat{\mathcal{L}}_H$

$\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{L}$

## ORCID iDs

Austin Rodriguez  0009-0000-3225-2673

Sakib Matin  0000-0001-5166-2838

Nicholas Lubbers  0000-0002-9001-9973

Kipton Barros  0000-0002-1333-5972

Jose L Mendoza-Cortes  0000-0001-5184-1406

## References

- [1] Unke O T, Chmiela S, Sauceda H E, Gastegger M, Poltavsky I, Schütt K T, Tkatchenko A and Müller K R 2021 *Chem. Rev.* **121** 10142–86
- [2] Behler J and Parrinello M 2007 *Phys. Rev. Lett.* **98** 146401
- [3] Smith J S, Isayev O and Roitberg A E 2017 *Chem. Sci.* **8** 3192–203
- [4] Smith J S, Isayev O and Roitberg A E 2017 *Sci. Data* **4** 170193
- [5] Smith J S, Nebgen B, Lubbers N, Isayev O and Roitberg A E 2018 *J. Chem. Phys.* **148** 241733
- [6] Smith J S, Zubatyuk R, Nebgen B, Lubbers N, Barros K, Roitberg A E, Isayev O and Tretiak S 2020 *Sci. Data* **7** 134
- [7] Lubbers N, Smith J S and Barros K 2018 *J. Chem. Phys.* **148** 241715
- [8] Zhang L, Han J, Wang H, Car R and E W 2018 *Phys. Rev. Lett.* **120** 143001
- [9] Bartók A P, Payne M C, Kondor R and Csányi G 2010 *Phys. Rev. Lett.* **104** 136403
- [10] Thompson A P, Swiler L P, Trott C R, Foiles S M and Tucker G J 2015 *J. Comput. Phys.* **285** 316–30
- [11] Shapeev A V 2016 *Multiscale Model. Simul.* **14** 1153–73
- [12] Schütt K T, Sauceda H E, Kindermans P J, Tkatchenko A and Müller K R 2018 *J. Chem. Phys.* **148** 241722
- [13] Batzner S, Musaelian A, Sun L, Geiger M, Mailoa J P, Kornbluth M, Molinari N, Smidt T E and Kozinsky B 2022 *Nat. Commun.* **13** 2453
- [14] Schütt K, Unke O and Gastegger M 2021 Equivariant message passing for the prediction of tensorial properties and molecular spectra *Int. Conf. on Machine Learning* pp 9377–88

- [15] Batatia I, Kovacs D P, Simm G, Ortner C and Csányi G 2022 *Adv. Neural Inf. Process. Syst.* **35** 11423–36
- [16] Chmiela S, Sauceda H E, Poltavsky I, Müller K R and Tkatchenko A 2019 *Comput. Phys. Commun.* **240** 38–45
- [17] Unke O T and Meuwly M 2019 *J. Chem. Theory Comput.* **15** 3678–93
- [18] Loew A, Sun D, Wang H-C, Botti S and Marques M A L 2025 *npj Comput. Mater.* **11** 178
- [19] Griewank A and Walther A 2008 *Evaluating Derivatives: Principles and Techniques of Algorithmic Differentiation* (SIAM)
- [20] Baydin A G, Pearlmutter B A, Radul A A and Siskind J M 2018 *J. Mach. Learn. Res.* **18** 1–43
- [21] Pulay P 2014 *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **4** 169–81
- [22] Pople J, Krishnan R, Schlegel H and Binkley J S 1979 *Int. J. Quantum Chem.* **16** 225–41
- [23] Gonzalez C and Schlegel H B 1989 *J. Chem. Phys.* **90** 2154–61
- [24] Fukui K 1981 *Acc. Chem. Res.* **14** 363–8
- [25] Rodriguez A, Smith J S and Mendoza-Cortes J L 2025 *J. Chem. Theory Comput.* **21** 6698–710
- [26] Yuan E C-Y, Kumar A, Guan X, Hermes E D, Rosen A S, Zádor J, Head-Gordon T and Blau S M 2024 *Nat. Commun.* **15** 8865
- [27] Amin I, Raja S and Krishnapriyan A 2025 Towards fast, specialized machine learning force fields: distilling foundation models via energy Hessians *The 13th Int. Conf. on Learning Representations ICLR 2025* (arXiv:2501.09009)
- [28] Herbold M and Behler J 2022 *J. Chem. Phys.* **156** 114106
- [29] Koker T, Gangan A, Kotak M, Marian J and Smidt T 2026 PFT: Phonon fine-tuning for machine learned interatomic potentials (arXiv:2601.07742)
- [30] Burger A, Thiede L, Rønne N, Bernales V, Vijaykumar N, Vegge T, Bhowmik A and Aspuru-Guzik A 2025 Shoot from the hip: Hessian interatomic potentials without derivatives (arXiv:2509.21624)
- [31] Williams N J, Kabalan L, Stojanovic L, Zólyomi V and Pyzer-Knapp E O 2025 *Sci. Data* **12** 9
- [32] Cui T, Han Y, Jia H, Duan C and Zhao Q 2026 *Sci. Data* **13** 37
- [33] Rodriguez A, Smith J S and Mendoza-Cortes J L 2025 OpenREACT-CHON-EFH — Open REaction Dataset of Atomic Configurations comprising C, H, O, N with Energies, Forces, and Hessians Figshare (<https://doi.org/10.6084/m9.figshare.29189858>)
- [34] Pearlmutter B A 1994 *Neural Comput.* **6** 147–60
- [35] Martens J et al 2010 Deep learning via hessian-free optimization *Proc. 27th Int. Conf. on Machine Learning* vol 27 pp 735–42
- [36] Nocedal J and Wright S J 2006 *Numerical Optimization* (Springer)
- [37] Halko N, Martinsson P G and Tropp J A 2011 *SIAM Rev.* **53** 217–88
- [38] Woodruff D P 2014 Sketching as a tool for numerical linear algebra (arXiv:1411.4357)
- [39] Hutchinson M 1990 *Commun. Stat. - Simul. Comput.* **19** 433–50
- [40] Avron H and Toledo S 2011 *J. ACM* **58** 17
- [41] Saibaba A K, Alexanderian A and Ipsen I C 2017 *Numer. Math.* **137** 353–95
- [42] Frisch M J et al 2016 Gaussian16 Revision C.01 gaussian Inc. Wallingford CT
- [43] Gao X, Ramezanghorbani F, Isayev O, Smith J S and Roitberg A E 2020 *J. Chem. Inf. Model.* **60** 3408–15
- [44] Devereux C, Smith J S, Huddleston K K, Barros K, Zubatyuk R, Isayev O and Roitberg A E 2020 *J. Chem. Theory Comput.* **16** 4192–202
- [45] Hardt M, Recht B and Singer Y 2016 Train faster, generalize better: stability of stochastic gradient descent *Proc. 33rd Int. Conf. on Machine Learning* vol 48 pp 1225–34
- [46] Hochreiter S and Schmidhuber J 1997 *Neural Comput.* **9** 1–42