MACHINE LEARNING FORCE FIELDS WITHOUT PRECOMPUTED LABELS: VARIATIONAL KOHN–SHAM TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Molecular dynamics at the accuracy of density functional theory (DFT) is limited by the cost of a self-consistent field (SCF) calculation at every step. Machine learning force fields (MLFFs) remove that cost from the simulation by regressing energies and forces onto reference DFT calculations, but move it to the dataset: every training label is itself a converged SCF calculation, and a new system, level of theory or range of conditions requires a new set of them. We merge labelling and training into a single optimisation. Because the Kohn–Sham energy is a variational functional of the electronic density, a network predicting the electronic state of a molecule can be trained by minimising the Kohn–Sham energy of its own prediction, averaged over geometries taken from an inexpensive empirical force field trajectory; the physics supplies the training signal and no labelled dataset is precomputed. The network emits a matrix in the atomic-orbital basis whose eigendecomposition yields an admissible density, energies follow from the functional and forces from the standard analytic nuclear gradient at that density, so inference requires no SCF iteration. Across ten drug-like and small organic molecules the model reaches a mean force error of 0.98 meV/Å against converged DFT, and on a temperature-extrapolation benchmark it reaches 5.6 meV/Å force RMSE far outside the training distribution, against 62.0 for the strongest supervised baseline. Against supervised training of the same network on the same geometries it is as accurate or better while using 12–17% less GPU time, and it drives stable molecular dynamics at a third of the cost per step of DFT.

1 INTRODUCTION

Molecular simulation of drug molecules and other organic systems relies on an accurate potential energy surface, and Kohn–Sham density functional theory (DFT) (Hohenberg & Kohn, 1964; Kohn & Sham, 1965) is one of the most widely used sources of such accuracy. Each DFT evaluation, however, requires solving a self-consistent field (SCF) problem whose cost grows steeply with system size, placing long molecular dynamics (MD) trajectories out of reach for all but the smallest systems. Machine learning force fields (MLFFs) are the standard remedy (Behler & Parrinello, 2007; Chmiela et al., 2017; Smith et al., 2017): by learning the mapping from atomic positions to energies and forces, modern equivariant architectures (Schütt et al., 2017; Batzner et al., 2022; Batatia et al., 2022; Musaelian et al., 2023) reach errors of a few meV/Å at a small fraction of the cost of DFT.

This efficiency at simulation time is bought with considerable expense at training time. Every training label is itself a converged DFT calculation, so the cost of DFT is not eliminated but shifted to the construction of the training set. Typical datasets contain tens of thousands to millions of labelled geometries (Smith et al., 2017; Chmiela et al., 2023), each scaling with system size and basis set, and more steeply for the hybrid and range-separated functionals often required for chemical accuracy. The cost also recurs: a new molecule, level of theory or region of configuration space requires a new round of labelling. Active learning (Podryabinkin & Shapeev, 2017; Smith et al., 2018) and transferable pre-trained models (Smith et al., 2017; Chmiela et al., 2023) reduce the number of labels needed but do not remove the need for them, and methods predicting the electronic structure directly (Unke et al., 2021; Li et al., 2022; Yu et al., 2023; Kim et al., 2026) require an entire converged Kohn–Sham Hamiltonian as the label for each geometry.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

We show that for Kohn–Sham DFT the labelling and training stages can be carried out as a single optimisation. The approach rests on a basic property of the theory: the Kohn–Sham energy is a variational functional of the electronic density, can be evaluated at any admissible density matrix from standard integrals, and attains its minimum at the ground-state density that solves the Kohn–Sham equations. A network predicting the density matrix of a molecule from its geometry can therefore be trained by minimising the Kohn–Sham energy of its own predictions, averaged over a distribution of geometries. This objective involves no converged energies, forces, densities or Hamiltonians, is bounded below by the ground-state energy, and requires only geometries, which we sample from an inexpensive empirical force field MD trajectory. Supervision is still present, supplied by the physics rather than by precomputed labels; we call this *variational training*. The network solves the same Kohn–Sham problem as conventional DFT, replacing the per-geometry SCF iterations with a minimisation amortised over all training geometries into one set of weights.

Turning this objective into a practical force field raises two challenges. The first is admissibility: the Kohn–Sham energy is meaningful only for densities that are idempotent and correctly normalised, which a network does not satisfy by default. We have it output an unconstrained symmetric matrix in the atomic-orbital basis and obtain the density through a generalised eigendecomposition followed by occupation of the lowest orbitals, a construction that is differentiable and admissible for every output. The second concerns forces: because the objective depends on the predicted matrix only through its occupied subspace, the trained network does not approximate the Fock operator, and forces read directly from its output, as is standard for supervised Hamiltonian predictors, are wrong by two orders of magnitude. We instead reconstruct the Fock matrix from the predicted density and evaluate the analytic nuclear gradient, which depends on the network only through what the objective constrains, so inference is one forward pass, one Fock build and one gradient evaluation, with no SCF iteration.

On standard drug molecule benchmarks the model attains a mean force error of 0.98 meV/Å with respect to converged DFT. On an out-of-distribution test at temperatures far above those of its training data, it matches the resolution of the published reference data on the in-distribution test sets and, at the highest test temperature, reaches a force RMSE of 5.6 meV/Å against 62.0 meV/Å for the strongest supervised MLFF. Since no labelled dataset is built in advance, the training distribution can be broadened simply by raising the sampling temperature at no additional cost, and a controlled comparison shows that the objective itself yields a threefold improvement over supervised baselines even under matched sampling. The model runs 1.9–$3.4\times$ faster than warm-started DFT.

Contributions.

- **Variational training** of MLFFs: an electronic-structure model trained by minimising the Kohn–Sham energy of its own predicted density, with no DFT labels and no SCF iteration at inference (Section 4).
- A **density parameterisation and force readout** for this objective, where the density is admissible by construction and forces come from a reconstructed Fock matrix depending only on the occupied subspace (Sections 4.2–4.3).
- **DFT-level accuracy and out-of-distribution performance**, with a controlled comparison separating what broader sampling contributes from what the objective itself does (Section 5).

2 Related Work

Machine learning force fields. MLFFs learn a mapping from atomic numbers and positions to energies and forces by regression onto the total energies and atomic forces DFT computes for a set of training geometries (Behler & Parrinello, 2007; Chmiela et al., 2017). Descriptor-based models have largely been superseded by message-passing networks (Schütt et al., 2017) and equivariant architectures such as PaiNN (Schütt et al., 2021), NequIP (Batzner et al., 2022), Allegro (Musaelian et al., 2023) and MACE (Batatia et al., 2022), which reach errors of a few meV/Å with modest amounts of data. Since every label requires a converged DFT calculation, much effort has gone into needing fewer: active learning (Podryabinkin & Shapeev, 2017; Smith et al., 2018), careful dataset construction (Christensen & von Lilienfeld, 2020; Chmiela et al., 2023) and pre-trained models meant to transfer across chemical space (Smith et al., 2017). All use a labelling budget more efficiently within a two-stage pipeline that labels a dataset first and fits a model afterwards; we merge the

two stages. Our forces also differ in kind: MLFFs differentiate a learned energy with respect to atomic positions, whereas we evaluate the analytic nuclear gradient of the Kohn–Sham energy at the predicted density.

Machine learning of electronic structure. A complementary line of work predicts electronic-structure quantities rather than energies and forces alone: wavefunctions (Schütt et al., 2019), symmetry-adapted atomic-orbital features (Qiao et al., 2020) and Kohn–Sham Hamiltonian matrices (Unke et al., 2021; Li et al., 2022; Yu et al., 2023), the last giving competitive energies and forces when inserted into standard quantum-chemistry routines (Kim et al., 2026). Trained to regress converged Hamiltonians, they need a full SCF calculation per geometry and a matrix-valued label. We adopt an architecture from this family but replace the regression loss with a variational one, which has consequences beyond the training data: the resulting matrix is not an approximation to the Fock operator, so their evaluation protocol no longer applies and a different force readout is required (Section 4.3).

Physics-informed objectives and SCF acceleration. Several works exploit the structure of DFT to reduce reliance on reference data. Brockherde et al. (2017) learn maps from the potential to the density and from the density to the energy, and Nagai et al. (2020) build physical constraints into learned exchange–correlation functionals. Closest in spirit is deep orbital-free DFT (Zhang et al., 2024), which also avoids SCF iterations, but by learning an approximate kinetic-energy functional and optimising the density per geometry at inference; we retain the exact Kohn–Sham kinetic energy and move the optimisation into training, amortising it across geometries in the sense of Amos (2023). Learned functionals (Kirkpatrick et al., 2021; Nagai et al., 2020) improve the solution the SCF procedure converges to without reducing the iteration count, and are compatible with our objective, which applies to any differentiable functional. A learned density or Hamiltonian is also often used as an SCF initial guess (Van Lenthe et al., 2006); our model can serve that role (Appendix I), but is meant to replace the SCF procedure rather than accelerate it.

3 Background

We consider closed-shell molecules in a finite Gaussian basis. Kohn–Sham DFT (Hohenberg & Kohn, 1964; Kohn & Sham, 1965) represents the electrons by N_{occ} doubly occupied orbitals $\psi_i = \sum_\mu C_{\mu i}\chi_\mu$ expanded in N atom-centred atomic orbitals (AOs). The AOs are not orthogonal, so the orbitals are orthonormal with respect to the overlap matrix $S_{\mu\nu}(R) = \langle\chi_\mu|\chi_\nu\rangle$, that is $C_{\mathrm{occ}}^\top S\, C_{\mathrm{occ}} = I$, and the electronic state is summarised by the density matrix $P = 2C_{\mathrm{occ}}C_{\mathrm{occ}}^\top$, which we call *admissible* when it arises in this way. The total energy is an explicit, differentiable function of P:

$$E_{\mathrm{KS}}[P] = \mathrm{Tr}\big(PH_{\mathrm{core}}\big) + \tfrac{1}{2}\mathrm{Tr}\big(PJ[P]\big) - \tfrac{\alpha}{4}\mathrm{Tr}\big(PK[P]\big) + E_{\mathrm{XC}}[\rho_P] + E_{\mathrm{nuc}}(R), \tag{1}$$

with H_{core} the one-electron integrals, $J[P]$ and $K[P]$ the Coulomb and exact-exchange matrices, α the fraction of exact exchange, E_{XC} the exchange–correlation functional evaluated by quadrature and E_{nuc} the nuclear repulsion. Every term can be evaluated for any admissible P, ground state or not.

The variational principle and SCF iterations. The ground-state energy $E_0(R)$ is the minimum of Eq. (1) over admissible densities, whose constraints $PSP = 2P$ and $\mathrm{Tr}(PS) = 2N_{\mathrm{occ}}$ depend on the geometry through $S = S(R)$. Stationarity gives the Kohn–Sham equation $F[P]\,C = S\,C\,\varepsilon$ with orbital energies ε and Fock matrix $F[P] = \partial E_{\mathrm{KS}}/\partial P$. Since F depends on the density it determines, the equation is nonlinear and is solved by self-consistent field (SCF) iteration: build $F[P]$, solve the generalised eigenvalue problem, form a new P from the N_{occ} lowest eigenvectors, and repeat until convergence. Even with acceleration (Pulay, 1980) and a good initial guess (Van Lenthe et al., 2006), this takes of order ten iterations, each a Fock build, and that repeated cost is what makes DFT expensive.

Analytic nuclear forces. The force on nucleus A is $-\mathrm{d}E_0/\mathrm{d}R_A$. Because the AOs move with the nuclei, the derivative contains, besides the Hellmann–Feynman term, a Pulay contribution from the geometry dependence of the basis (Pulay, 1969; Helgaker & Jørgensen, 1988):

$$\frac{\mathrm{d}E}{\mathrm{d}R_A} = \mathrm{Tr}\left(P\,\frac{\partial H_{\mathrm{core}}}{\partial R_A}\right) + \left(\frac{\partial(E_2 + E_{\mathrm{XC}})}{\partial R_A}\right)_P - \mathrm{Tr}\left(W\,\frac{\partial S}{\partial R_A}\right) + \frac{\partial E_{\mathrm{nuc}}}{\partial R_A}, \tag{2}$$

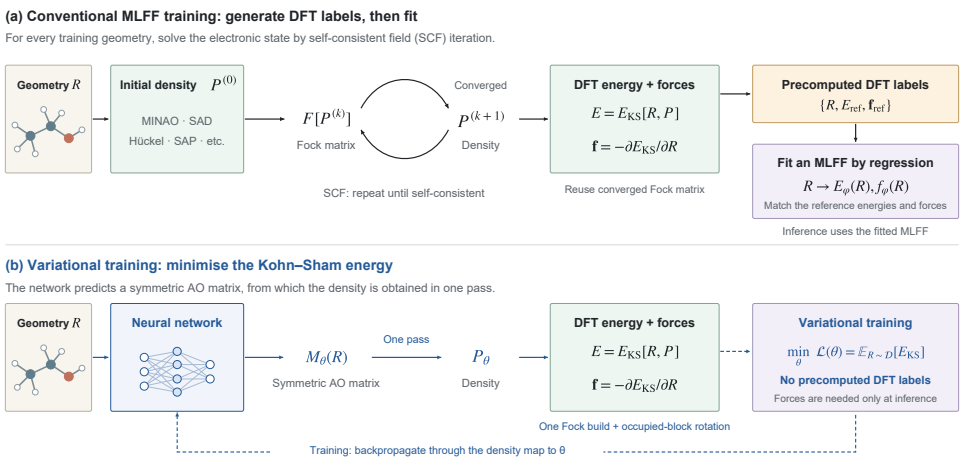

Figure 1: **Conventional and variational training of a force field. (a)** A conventional MLFF is fit to precomputed DFT labels, so every training geometry requires a converged SCF calculation: from an initial guess $P^{(0)}$, each iteration builds $F[P^{(k)}]$, solves $F[P^{(k)}]C = SC\varepsilon$ and occupies the N_{occ} lowest orbitals to obtain the next density, until the density stops changing; the resulting energies and forces are then stored and regressed onto. **(b)** We keep the density construction and the DFT energy expression but replace the per-geometry iteration by a network emitting an unconstrained symmetric matrix $M_\theta(R)$, diagonalised and occupied exactly as above but only once, so the density of Eq. (4) follows in a single pass with no Fock rebuild and no convergence test. Parameters are trained by minimising the Kohn–Sham energy of that density (Eq. (3)), with gradients flowing back through the density map, on geometries from classical MD and with no DFT label precomputed. Forces are needed only at inference, where one Fock build and an occupied-block rotation precede the standard analytic gradient.

where E_2 is the two-electron energy, $(\cdot)_P$ denotes differentiation at fixed P, and $W = 2\sum_{i\le N_{\mathrm{occ}}} \varepsilon_i\, C_{\cdot i} C_{\cdot i}^\top$ is the energy-weighted density matrix, equal to $\frac{1}{2}PFP$ for orbitals satisfying the Kohn–Sham equation. Evaluating it requires no iteration, but W needs orbital energies from a Fock matrix consistent with P, so an accurate density alone is not sufficient (Section 4.3).

4 Method

Our method consists of four components, summarised in Figure 1: a variational training objective (Section 4.1), a parameterisation that maps the network output to an admissible density (Section 4.2), a procedure for computing energies and forces from that density (Section 4.3), and the architecture, the source of training geometries and an efficient inference implementation (Section 4.4).

4.1 Variational training objective

Let $\mathcal{D}$ be a distribution over molecular geometries and $P_\theta(R)$ the density matrix predicted by a network with parameters θ. We train it by minimising the expected Kohn–Sham energy of its predictions,

$$\mathcal{L}(\theta) \;=\; \mathbb{E}_{R\sim\mathcal{D}}\big[\, E_{\mathrm{KS}}\big(R,\, P_\theta(R)\big)\,\big]. \tag{3}$$

Three properties make this objective suitable for training. It can be computed from the nuclear coordinates and standard integrals alone, without any precomputed target. Provided $P_\theta(R)$ is admissible, it is bounded below by $\mathbb{E}_R[E_0(R)]$ by the variational principle, with equality only when the prediction is the ground-state density for almost every geometry drawn from $\mathcal{D}$, so its value measures the remaining variational error directly. And it is differentiable with respect to P, so exact gradients with respect to θ follow by automatic differentiation. With sufficient capacity and a global minimiser, the network recovers the ground-state density for almost every geometry drawn from $\mathcal{D}$.

Each step evaluates the Kohn–Sham functional at the model's output and descends on it, so solving the variational problem for the training geometries is the same work that labelling would otherwise

perform; what differs is how that work is organised. A conventional pipeline converges an SCF calculation per geometry, stores the result, and fits a model to it in a separate stage. Here the variational problems for all training geometries are solved jointly through a shared set of weights, inside the optimisation that trains the model, and no converged energy, force, density or Hamiltonian is stored or regressed onto, so the set of training geometries can be changed without a new labelling stage. We call this *variational training*.

4.2 From an unconstrained matrix to an admissible density

Eq. (3) is meaningful only for admissible density matrices, which must be symmetric, S-idempotent and of the correct trace. Rather than encouraging these constraints with a penalty, we satisfy them by construction, using the same parameterisation as the SCF procedure. The network outputs an unconstrained symmetric matrix $M_\theta(R) \in \mathbb{R}^{N\times N}$. Using the Cholesky factorisation $S = LL^\top$ and $X = L^{-\top}$, we diagonalise M_θ in the resulting orthonormal basis and occupy the N_{occ} lowest eigenvectors:

$$X^\top M_\theta X = U\varepsilon U^\top, \qquad C_{\mathrm{occ}} = X\,U_{:,1:N_{\mathrm{occ}}}, \qquad P_\theta(R) = 2\,C_{\mathrm{occ}}C_{\mathrm{occ}}^\top. \tag{4}$$

Since $X^\top SX = I$, the columns of C_{occ} are S-orthonormal for any M_θ, and P_θ is therefore admissible for every network output. The map is differentiable, so gradients of Eq. (3) propagate to M_θ and from there to θ.

Stable backpropagation through the eigendecomposition. The generic backward pass of a symmetric eigendecomposition contains reciprocal gaps between all pairs of eigenvalues, including the nearly degenerate core levels of a molecule, so it overflows and training diverges within a few steps. We therefore differentiate the occupied projector $\Pi = U_{:,1:N_{\mathrm{occ}}}U_{:,1:N_{\mathrm{occ}}}^\top$ directly, with a custom backward pass, Eq. (6) in Appendix G, whose only denominators are occupied–virtual gaps. It also shows that the loss reaches M_θ only through the mixing between occupied and virtual eigenvectors: its eigenvalues, diagonal blocks and overall scale receive no gradient. The trained M_θ therefore determines an accurate occupied subspace, and hence an accurate density, but is not an approximation to the Fock operator, which determines how we compute forces.

4.3 Energies and forces without SCF iterations

Energy, and why forces cannot be read from the network output. The energy follows from evaluating Eq. (1) at $P_\theta(R)$ at the level of theory used for training, and is variational: because $P_\theta(R)$ is admissible, it never falls below the converged ground-state energy. Forces are harder. The analytic gradient in Eq. (2) requires orbital energies to form W, and supervised Hamiltonian predictors obtain these by diagonalising the predicted matrix and passing its eigenvalues and eigenvectors to a standard gradient routine. That protocol fails here, because the eigenvalues of M_θ are never trained (Section 4.2): on aspirin it gives a force error of roughly 0.5 eV/Å, two orders of magnitude above what the same model's density supports (Appendix H).

Fock reconstruction and canonicalisation. We therefore use M_θ only to produce the density in Eq. (4) and reconstruct the Fock matrix $F = F[P_\theta] = \partial E_{\mathrm{KS}}/\partial P$ at that density, which requires one Fock build. The predicted occupied orbitals are in general not eigenvectors of F, so we rotate them within the occupied subspace before forming W:

$$F_{oo} = C_{\mathrm{occ}}^\top F\,C_{\mathrm{occ}} = VeV^\top, \qquad C'_{\mathrm{occ}} = C_{\mathrm{occ}}V, \qquad \varepsilon_i = e_i. \tag{5}$$

The resulting energy-weighted density matrix equals $W = \frac{1}{2}PF[P]P$, and we pass $(\varepsilon, C'_{\mathrm{occ}}, \mathrm{occ})$ to a standard analytic gradient routine including the Pulay and grid-response terms. The extra cost is one $N_{\mathrm{occ}} \times N_{\mathrm{occ}}$ eigendecomposition, under a millisecond. The rotation is necessary: forming W from the diagonal of F_{oo} discards its off-diagonal block and introduces a corresponding error in the Pulay term, raising the force error by two orders of magnitude (Appendix H).

Consistency between readout and objective. Eqs. (4)–(5) depend on M_θ only through its occupied subspace: perturbing its diagonal blocks leaves P, $F[P]$, W and the forces unchanged to numerical precision, whereas occupied–virtual perturbations change them. That is exactly the subspace to which Eq. (6) propagates gradients, so every reported quantity depends only on parts of the network output

that training constrains. Inference is therefore one forward pass, one Fock build and one gradient evaluation; we take no additional Roothaan step, whose undamped update overshoots for several of our molecules (Appendix H).

4.4 Architecture, training geometries and efficient inference

Architecture and training geometries. The map $R \mapsto M_\theta(R)$ must be equivariant: rotating a molecule transforms an AO-basis matrix blockwise by the Wigner-D matrices of the corresponding angular momenta. We adopt the QHFlow2 backbone (Kim et al., 2026), an eSCN-style network with SO(2)-reduced convolutions (Passaro & Zitnick, 2023), in its released small configuration with 13.7M parameters and without architectural changes. The objective requires only geometries, which we draw from classical MD under an empirical force field. That force field is a proposal distribution alone: its energies and forces are never used, so it must sample broadly rather than accurately, and since the training cost is set by the number of optimisation steps rather than by the origin of the geometries, its sampling conditions can be changed without affecting what training costs (Section 5.4).

Training pipeline. Evaluating Eq. (3) for a geometry requires quantities that do not depend on the network, such as the density-fitting tensor, the atomic orbitals on the integration grid and the minimal-basis anchor, whose construction takes considerably longer than a gradient step. We build them in parallel worker processes that overlap the gradient computation, and, since preparation is the bottleneck, use each prepared geometry for several steps by drawing minibatches without replacement from a pool that is refreshed as training proceeds (Appendix A).

Efficient inference. Two features of the inference procedure admit optimisations that a conventional DFT calculation cannot use: it performs one Fock build and one gradient evaluation, so numerical error enters once rather than being fed back over SCF iterations, and the forces need only be accurate relative to the model's own error. We therefore share integral screening and setup across routines, build the exchange–correlation contribution only in the occupied–occupied block that Eq. (5) and $W = \frac{1}{2}PFP$ require, and run the quadratures, weights, screening and the eigendecomposition of Eq. (4) in single precision with two double-precision refinement sweeps, keeping in double precision any computation whose accumulated magnitude times the relative round-off of single precision would exceed its output tolerance (Appendix B). We implement this for PBE/def2-SVP; on molecules of 9 to 42 atoms it increases the force error by less than 2% of the model's own error.

5 Experiments

We evaluate accuracy against DFT on rMD17, compare with supervised training of the same network, and test on 3BPA how the model extrapolates to temperatures far above those sampled in training. Two further experiments are reported in the appendices: the model drives molecular dynamics over 100 ps with an energy drift not resolved from that of Born–Oppenheimer MD, at a third of the cost per step (Appendix D), and a single model trained across eight ligands of a protein target reaches a force MAE of 0.93 meV/Å at 52 to 64 atoms (Appendix E).

5.1 Experimental setup

Benchmarks and protocol. rMD17 (Christensen & von Lilienfeld, 2020) contains ten drug-like and small organic molecules and is the standard benchmark for molecule-specific MLFFs; we use its level of theory, PBE/def2-SVP (Perdew et al., 1996; Weigend & Ahlrichs, 2005), with density fitting. 3BPA (Kovács et al., 2021) is a temperature-extrapolation benchmark at ωB97X/6-31G(d) (Chai & Head-Gordon, 2008), on which baselines are trained at 300 K and every model is tested at 300, 600 and 1200 K and on dihedral scans. Appendix D uses the 42-atom tetrapeptide Ac-Ala$_3$-NHMe from MD22 (Chmiela et al., 2023). Both benchmarks are evaluated on their complete test sets, which are not used in training, in the metric the baselines report: MAE on rMD17 ($1{,}000$ frames per molecule) and RMSE on 3BPA ($1{,}669$, $2{,}138$ and $2{,}139$ frames at 300, 600 and 1200 K and $7{,}047$ across the dihedral scans). Energy errors against the labels are computed after removing the mean signed error over each test set, since the baselines absorb such an offset in their fitted per-atom reference energies. The efficient inference implementation of Section 4.4 is built for PBE/def2-SVP, so the 3BPA results use the standard double-precision path.

Table 1: **Accuracy on rMD17.** Energy (E, meV) and force (F, meV/Å) MAE against the rMD17 labels. Baselines are trained on 950 labelled configurations; our two models use no precomputed labels and differ only in the thermostat temperature of the trajectory supplying their training geometries. Energies are compared after removing the constant offset between the two programs. The last column scores our own converged SCF against the same labels. Best model per molecule and quantity in bold, ties all marked.

	NequIP		Allegro		BOTNet		MACE		Variational				*Conv. DFT*	
									500 K		1600 K			
Molecule	E	F	E	F	E	F	E	F	E	F	E	F	E	F
Aspirin	2.3	8.2	2.3	7.3	2.3	8.5	2.2	6.6	0.8	2.3	**0.2**	**1.9**	*0.2*	*0.5*
Azobenzene	0.7	2.9	1.2	2.6	0.7	3.3	1.2	3.0	0.2	1.5	**0.1**	**1.0**	*0.1*	*0.3*
Benzene	0.04	0.3	0.3	**0.2**	**0.03**	0.3	0.4	0.3	0.1	0.3	0.1	0.4	*0.1*	*0.3*
Ethanol	0.4	2.8	0.4	2.1	0.4	3.2	0.4	2.1	**0.1**	0.9	**0.1**	**0.6**	*0.1*	*0.2*
Malonaldehyde	0.8	5.1	0.6	3.6	0.8	5.8	0.8	4.1	0.9	5.0	**0.1**	**1.3**	*0.1*	*0.6*
Naphthalene	0.9	1.3	**0.2**	0.9	**0.2**	1.8	0.5	1.6	**0.2**	0.7	**0.2**	**0.5**	*0.2*	*0.3*
Paracetamol	1.4	5.9	1.5	4.9	1.3	5.8	1.3	4.8	0.3	1.8	**0.2**	**1.3**	*0.1*	*0.4*
Salicylic acid	0.7	4.0	0.9	2.9	0.8	4.3	0.9	3.1	0.8	2.1	**0.2**	**1.8**	*0.1*	*0.5*
Toluene	0.3	1.6	0.4	1.8	0.3	1.9	0.5	1.5	**0.1**	0.6	**0.1**	**0.5**	*0.1*	*0.2*
Uracil	0.4	3.1	0.6	1.8	0.4	3.2	0.5	2.1	1.8	3.9	**0.1**	**1.7**	*0.1*	*0.5*

References. The labels were produced by a different program, with different integration and integral settings, so our own converged SCF does not reproduce them exactly. We therefore also report the error of that converged calculation against the same labels, written *Conv. DFT* in the tables: it measures how closely two independent DFT implementations agree, and sets the scale below which a label-scored comparison stops being informative. Errors against our own converged SCF are in Appendix C.

Training geometries and baselines. For each molecule we run a 100 ns classical MD simulation under the GAFF2 force field (Wang et al., 2004) and draw 95,000 training and 5,000 validation conformations as disjoint subsets of a random permutation of its frames, at a thermostat temperature of 1600 K unless stated otherwise; remaining settings are in Appendix A. We compare against equivariant MLFFs trained by regression on the benchmark labels: on rMD17, NequIP (Batzner et al., 2022), Allegro (Musaelian et al., 2023), BOTNet and MACE (Batatia et al., 2022), each trained on 950 labelled configurations with 50 for validation, as reported by Batatia et al. (2022); on 3BPA, linear ACE (Kovács et al., 2021), Allegro, NequIP, BOTNet and MACE, each trained on 450 labelled configurations sampled at 300 K with 50 for validation, as reported in the respective papers. We additionally train the same network we use with the supervised objective it was designed for, on the same geometries, which isolates the effect of the training signal from that of the architecture (Section 5.3). We train one model per molecule and per sampling temperature with a single seed, and every comparison between our own models holds the test frames fixed and is therefore paired. Across all evaluations reported here, the density map returned an admissible density and finite forces on every frame.

5.2 Accuracy on rMD17

Table 1 compares variational training with supervised MLFFs. The force error of the model trained on 1600 K geometries is lower than the best baseline's on nine of the ten molecules, by up to a factor of 3.6, and its energy error is lower or equal on nine of ten, by up to a factor of ten. The exception in both cases is benzene, where the baselines already agree with the labels as closely as an independent converged DFT calculation does; elsewhere that agreement is much closer than the error of any model, so the comparison is between models rather than DFT implementations. The energy errors are at most 0.23 meV on every molecule, and on six of the ten they equal those of converged DFT to within 0.01 meV, so at this precision the labels cannot distinguish the predictions from a converged calculation.

The sampling temperature of the training geometries matters: sampled at 500 K instead of 1600 K, the mean force error rises by 75% and nine of the ten molecules become worse. The largest effect

Table 2: **Variational versus supervised training of the same network**, on four rMD17 molecules of 9 to 24 atoms. Both arms share the 95,000 training geometries, network, optimiser, schedule, seed and GPU type, and are scored on identical frames against our own converged SCF at PBE/def2-SVP. Supervised training regresses onto converged Fock matrices for those geometries; its cost is labelling plus training.

	Force MAE (meV/Å)		Energy MAE (meV)		Total training cost (GPU-h)			
					supervised			
Molecule	supervised	variational	supervised	variational	labelling	training	total	variational
Ethanol	0.57	**0.51**	0.012	**0.008**	7.7	13.0	20.6	**17.6**
Malonaldehyde	1.17	**1.12**	0.035	**0.034**	9.5	10.6	20.1	**17.0**
Aspirin	2.23	**1.78**	0.637	**0.149**	40.6	10.1	50.8	**42.2**
Azobenzene	1.23	**0.94**	0.360	**0.080**	40.3	10.5	50.8	**44.9**

Table 3: **Temperature extrapolation on 3BPA.** Energy (E, meV) and force (F, meV/Å) RMSE against the 3BPA labels at ωB97X/6-31G(d). Baselines are trained on 450 labelled configurations sampled at 300 K; the variationally trained models use no precomputed labels and differ only in the temperature at which their training geometries were sampled, 300 K matching the baselines and 1600 K the setting we propose. The dihedral row pools the three scans, and energies are compared after removing the constant offset between the two programs. The last column scores our own converged SCF against the same labels; the best model in each row is in bold.

							Variational	
Test set		ACE	Allegro	NequIP	BOTNet	MACE	300 K	1600 K	*Conv. DFT*
300 K	E	7.1	3.8	3.3	3.1	3.0	**1.3**	**1.3**	*1.3*
	F	27.1	13.0	10.8	11.0	8.8	5.6	**5.2**	*5.2*
600 K	E	24.0	12.1	11.2	11.5	9.7	1.8	**1.6**	*1.6*
	F	64.3	29.1	26.4	26.7	21.8	6.7	**5.3**	*5.2*
1200 K	E	85.3	42.6	38.5	39.1	29.8	15.4	**2.0**	*2.0*
	F	187.0	83.0	76.2	81.1	62.0	20.8	**5.6**	*5.2*
Dihedral	E	22.2	—	—	16.3	7.8	0.7	**0.6**	*0.7*
	F	39.2	—	—	20.0	16.5	6.0	**5.3**	*5.4*

is on malonaldehyde, whose GAFF2 torsion parameters restrict the 500 K sampling so that a third of the test frames lie outside the training distribution.

5.3 Comparison with supervised training of the same network

The comparisons above are with MLFFs of a different architecture. To isolate the effect of the training signal, we train the same network on the same geometries by the supervised route it was designed for: converged Fock matrices are computed for every training geometry and the network is fit to them with the Hamiltonian loss of Kim et al. (2026). Everything else, including the frame order, optimiser, schedule, seed and GPU type, is held fixed.

Table 2 reports accuracy and cost for four molecules spanning the size range of rMD17. Variational training gives lower force and energy errors on all four, by 5 to 24% on forces and up to a factor of 4.5 on energies. As initial guesses for a subsequent SCF calculation the two are equivalent, within 0.12 iterations; they differ in the single-pass energies and forces. The end-to-end cost favours variational training by 12–17%: supervised *training* is faster, evaluating no integrals in the optimisation loop, but the labels it needs cost more than that saving, and that term grows with system size while the training term does not. A GPU-occupancy process ran alongside the variational runs but not the supervised ones, which if anything overstates our cost. The same comparison at the hybrid level of 3BPA (Appendix F) costs the same to within 3% on both routes, since labels for a 27-atom molecule in a small basis are cheap, and there our force error is half the supervised one on all six test sets.

5.4 TEMPERATURE EXTRAPOLATION ON 3BPA

Table 3 reports the results. Variational training reduces the force RMSE at 1200 K by an order of magnitude relative to MACE, the strongest baseline, and the energy RMSE by a factor of fifteen, and its error grows little with the test temperature: against our own converged SCF the force RMSE increases by a factor of 1.7 from the 300 K to the 1200 K test set (Appendix C). The two DFT implementations agree far less closely on 3BPA than on rMD17, and at 300 K, 600 K and on the dihedral scans the model matches that level of agreement, so there the label-scored metric reflects the precision of the labels rather than the model's error; even at 1200 K its force RMSE lies only 0.4 meV/Å above that of converged DFT, against 57 meV/Å for MACE. The limited agreement is label-side: finite differences of our energy reproduce our analytic forces to 1.8×10^{-4} meV/Å, whereas the labels differ from them by 6.55 meV/Å.

The control trained at 300 K, the temperature used by the baselines, errs nearly four times more at 1200 K than at 300 K; against our own converged SCF its force RMSE grows by a factor of 10.7 across those test sets, comparable to about seven for the supervised baselines, against 1.7 for the 1600 K model. Since the two differ only in sampling temperature, the breadth of the training geometries accounts for most of the robustness under extrapolation, and costs nothing but a change of thermostat. What the 300 K trajectory lacks is not primarily torsional: at 1200 K its error is as large in torsion-space cells it visited often as in cells it never visited, so the missing coverage lies elsewhere, plausibly in thermal distortion of bond lengths and angles. Even with its training geometries sampled at the baselines' own temperature, variational training remains about three times more accurate at 1200 K than MACE.

6 DISCUSSION AND LIMITATIONS

The cost profile is that of a quantum-chemistry code. Inference is a network forward pass followed by a Fock build and an analytic nuclear gradient, both full-basis operations, so removing the self-consistent loop leaves a small constant speedup over converged DFT — 1.9 to 3.4× against a warm start in our measurements — rather than the orders of magnitude a fitted potential delivers. The method is an alternative to supervised MLFFs in accuracy and in independence from precomputed data, not in throughput per geometry.

The proposal distribution is a dependency. Training geometries come from a classical force field, which must exist for the system and reach the regions the model is expected to describe. Malonaldehyde shows the failure mode: GAFF2's torsion parameters confine its 500 K sampling so narrowly that a third of the test frames fall outside the training distribution, the largest 500 K force error we observe. Raising the temperature repairs it here, but coverage inherits the defects of the proposal in general.

7 CONCLUSION

A machine learning force field does not require a precomputed dataset of reference calculations. Training a network to minimise the variational Kohn–Sham energy of its own predicted density merges labelling and training into a single optimisation, and because the unconstrained network output is made admissible by an eigendecomposition rather than by a penalty, the predicted density is valid for every input at every point in training. The resulting models reach a mean force error of 0.98 meV/Å against converged DFT on rMD17 and, on 3BPA, stay at the resolution of the benchmark's own labels wherever that limit binds, while remaining an order of magnitude more accurate than the strongest supervised baseline where it does not. Two consequences seem worth emphasizing: the sampling distribution becomes a free design variable, since no reference calculations follow from the choice of training geometries and our controlled comparison attributes most of the robustness under extrapolation to that breadth; and merging the two stages does not cost compute, taking 12–17% less GPU time than the supervised route on rMD17 and the same on 3BPA. Once a force field no longer depends on a labelled dataset, where to train it stops being a question of labelling budget.

AI USE STATEMENT

In this work, we used generative AI tools to help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, and interpret results. Generating synthetic data sets, assisting with translation, cleaning and reformatting datasets, and supporting qualitative and thematic data analysis are not applicable to this work. Additionally, we used generative AI tools to create or modify scientific figures or images, suggest experimental parameters, create or edit software code, draft parts of a research paper, summarise or analyse existing literature, discover research topics or identify gaps, brainstorm, source or search for information, edit a research paper to improve readability, identify relevant literature, format references, suggest a structure for a research paper, and propose a title or keywords for a research paper. We have reviewed all AI-assisted work. LLM-generated code was verified and tested by the authors before the runs it produced were used, every number reported here is the output of an executed run rather than a model's account of one, and every reference was checked against the publisher record. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

The components a reader would need to reimplement this work are stated in the paper itself. The objective is Eq. (3), the density map Eq. (4) and the force readout Section 4.3; the backward pass through the eigendecomposition is derived in Appendix G and the readout is compared against the alternatives we tried in Appendix H. The full training configuration — level of theory, bases, integration grid, tolerances, conformation counts, optimiser, schedule and the precision used in each part of the model — is collected in one table in Appendix A, and the training pipeline and the inference implementation are described in Appendix B. Section 5.1 states how the training geometries are generated, which test splits and metrics are used, and where the baseline numbers come from; since training consumes geometries alone, no label set has to be distributed for a training set to be reconstructed. Appendix C reports every accuracy result a second time against our own converged SCF, which removes differences between quantum-chemistry programs from the comparison, and Appendix F gives the protocol for the supervised model we train ourselves. Source code, configuration files and the job launchers for each experiment are withheld during review for anonymity and will be released with the paper.

REFERENCES

Brandon Amos. Tutorial on amortized optimization. *Foundations and Trends in Machine Learning*, 16(5):592–732, 2023.

Ilyes Batatia, Dávid Péter Kovács, Gregor N. C. Simm, Christoph Ortner, and Gábor Csányi. MACE: Higher order equivariant message passing neural networks for fast and accurate force fields. In *Advances in Neural Information Processing Systems*, volume 35, pp. 11423–11436, 2022.

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):2453, 2022.

Jörg Behler and Michele Parrinello. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Physical Review Letters*, 98(14):146401, 2007.

Felix Brockherde, Leslie Vogt, Li Li, Mark E Tuckerman, Kieron Burke, and Klaus-Robert Müller. Bypassing the Kohn-Sham equations with machine learning. *Nature Communications*, 8(1):872, 2017.

Jeng-Da Chai and Martin Head-Gordon. Systematic optimization of long-range corrected hybrid density functionals. *The Journal of Chemical Physics*, 128(8):084106, 2008.

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):e1603015, 2017.

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. *Science Advances*, 9(2):eadf0873, 2023.

Anders S. Christensen and O. Anatole von Lilienfeld. On the role of gradients for machine learning of molecular energies and forces. *Machine Learning: Science and Technology*, 1(4):045018, 2020.

B. I. Dunlap. Fitting the coulomb potential variationally in Xα molecular calculations. *The Journal of Chemical Physics*, 78(6):3140–3142, 1983.

David F Hahn, Christopher I Bayly, Melissa L Boby, Hannah E. Bruce Macdonald, John D Chodera, Vytautas Gapsys, Antonia S J S Mey, David L Mobley, Laura Perez Benito, Christina E M Schindler, Gary Tresadern, and Gregory L Warren. Best practices for constructing, preparing, and evaluating protein-ligand binding affinity benchmarks [article v1.0]. *Living Journal of Computational Molecular Science*, 4(1):1497, 2022. doi: 10.33011/livecoms.4.1.1497.

Trygve Helgaker and Poul Jørgensen. Analytical calculation of geometrical derivatives in molecular electronic structure theory. *Advances in Quantum Chemistry*, 19:183–245, 1988.

Pierre Hohenberg and Walter Kohn. Inhomogeneous electron gas. *Physical Review*, 136(3B): B864–B871, 1964.

Seongsu Kim, Chanhui Lee, Yoonho Kim, Seongjun Yun, Honghui Kim, Nayoung Kim, Changyoung Park, Sehui Han, Sungbin Lim, and Sungsoo Ahn. Machine learning Hamiltonians are accurate energy-force predictors. *arXiv preprint arXiv:2602.16897*, 2026. URL https://arxiv.org/abs/2602.16897.

James Kirkpatrick, Brendan McMorrow, David H P Turban, Alexander L Gaunt, James S Spencer, Alexander G D G Matthews, Annette Obika, Louis Thiry, Meire Fortunato, David Pfau, et al. Pushing the frontiers of density functionals by solving the fractional electron problem. *Science*, 374(6573):1385–1389, 2021.

Walter Kohn and Lu Jeu Sham. Self-consistent equations including exchange and correlation effects. *Physical Review*, 140(4A):A1133–A1138, 1965.

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.

He Li, Zun Wang, Nianlong Zou, Meng Ye, Runzhang Xu, Xiaoxun Gong, Wenhui Duan, and Yong Xu. Deep-learning density functional theory Hamiltonian for efficient ab initio electronic-structure calculation. *Nature Computational Science*, 2(6):367–377, 2022.

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. *Nature Communications*, 14(1):579, 2023.

Ryo Nagai, Ryosuke Akashi, and Osamu Sugino. Completing density functional theory by machine learning hidden messages from molecules. *npj Computational Materials*, 6(1):43, 2020.

Saro Passaro and C Lawrence Zitnick. Reducing SO(3) convolutions to SO(2) for efficient equivariant GNNs. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 27420–27438. PMLR, 2023.

John P Perdew, Kieron Burke, and Matthias Ernzerhof. Generalized gradient approximation made simple. *Physical Review Letters*, 77(18):3865–3868, 1996.

Evgeny V Podryabinkin and Alexander V Shapeev. Active learning of linearly parametrized interatomic potentials. *Computational Materials Science*, 140:171–180, 2017.

Peter Pulay. Ab initio calculation of force constants and equilibrium geometries in polyatomic molecules: I. Theory. *Molecular Physics*, 17(2):197–204, 1969.

Péter Pulay. Convergence acceleration of iterative sequences. the case of SCF iteration. *Chemical Physics Letters*, 73(2):393–398, 1980.

Zhuoran Qiao, Matthew Welborn, Animashree Anandkumar, Frederick R. Manby, and Thomas F. Miller, III. OrbNet: Deep learning for quantum chemistry using symmetry-adapted atomic-orbital features. *The Journal of Chemical Physics*, 153(12):124111, 2020.

Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Sauceda Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Kristof Schütt, Oliver Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9377–9388. PMLR, 2021.

Kristof T Schütt, Michael Gastegger, Alexandre Tkatchenko, K-R Müller, and Reinhard J Maurer. Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nature Communications*, 10(1):5024, 2019.

Justin S Smith, Olexandr Isayev, and Adrian E Roitberg. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chemical Science*, 8(4):3192–3203, 2017.

Justin S Smith, Ben Nebgen, Nicholas Lubbers, Olexandr Isayev, and Adrian E Roitberg. Less is more: Sampling chemical space with active learning. *The Journal of Chemical Physics*, 148(24): 241733, 2018.

Qiming Sun, Xing Zhang, Samragni Banerjee, Peng Bao, Marc Barbry, Nick S Blunt, Nikolay A Bogdanov, George H Booth, Jia Chen, Zhi-Hao Cui, et al. Recent developments in the PySCF program package. *The Journal of Chemical Physics*, 153(2):024109, 2020. doi: 10.1063/5.0006074.

Oliver T Unke, Mihail Bogojeski, Michael Gastegger, Mario Geiger, Tess Smidt, and Klaus-Robert Müller. SE(3)-equivariant prediction of molecular wavefunctions and electronic densities. In *Advances in Neural Information Processing Systems*, volume 34, pp. 14434–14447, 2021.

O. Vahtras, J. Almlöf, and M. W. Feyereisen. Integral approximations for LCAO-SCF calculations. *Chemical Physics Letters*, 213(5–6):514–518, 1993.

J. H. Van Lenthe, Renate Zwaans, Huub J. J. Van Dam, and M. F. Guest. Starting SCF calculations by superposition of atomic densities. *Journal of Computational Chemistry*, 27(8):926–932, 2006.

Junmei Wang, Romain M Wolf, James W Caldwell, Peter A Kollman, and David A Case. Development and testing of a general amber force field. *Journal of Computational Chemistry*, 25(9): 1157–1174, 2004.

Florian Weigend and Reinhart Ahlrichs. Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy. *Physical Chemistry Chemical Physics*, 7(18):3297–3305, 2005.

Jerry L Whitten. Coulombic potential energy integrals and approximations. *The Journal of Chemical Physics*, 58(10):4496–4501, 1973.

Xiaojie Wu, Qiming Sun, Zhichen Pu, Tianze Zheng, Wenzhi Ma, Wen Yan, Yu Xia, Zhengxiao Wu, Mian Huo, Xiang Li, Weiluo Ren, Sheng Gong, Yumin Zhang, and Weihao Gao. Enhancing GPU-acceleration in the Python-based simulations of chemistry frameworks. *WIREs Computational Molecular Science*, 15(2):e70008, 2025. doi: 10.1002/wcms.70008.

Haiyang Yu, Zhao Xu, Xiaofeng Qian, Xiaoning Qian, and Shuiwang Ji. Efficient and equivariant graph networks for predicting quantum Hamiltonian. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 40412–40424. PMLR, 2023.

He Zhang, Siyuan Liu, Jiacheng You, Chang Liu, Shuxin Zheng, Ziheng Lu, Tong Wang, Nanning Zheng, and Bin Shao. Overcoming the barrier of orbital-free density functional theory for molecular systems using deep learning. *Nature Computational Science*, 4(3):210–223, 2024.

A TRAINING CONFIGURATION

Table A1 lists the architecture, level of theory, training data, optimiser settings and numerical precision of every model in this paper. The multi-ligand model of Appendix E follows the 3BPA column with an added sulfur basis, ωB97X/6-31G(d)+S($4s3p1d$), and is trained on geometries of all eight ligands drawn from classical trajectories at 300 K.

The per-step cost is dominated by the `float64` physics layer rather than by the network: the Kohn–Sham energy evaluation and its backward pass are essentially independent of model size and scale with the molecule instead, so the 13.7M-parameter model trains at nearly the same rate as a 42.8M-parameter one. CPU-side integral construction (density-fitting tensors, Becke grid AO values) runs in worker processes and is overlapped with optimisation, as described in Section 4.4.

Table A1: Training configuration. Both benchmarks use the same architecture, training-data protocol and optimiser; the level of theory and the auxiliary basis differ.

	rMD17 / MD22	**3BPA**
Model		
Backbone	QHFlow2 (Kim et al., 2026), released *small* configuration	
Parameters	13.7M	
Hidden width / bottleneck	64 / 32	
Message-passing layers	4	
Hamiltonian-refinement layers	2	
Interaction radius	15 Bohr	
Max. neighbours per atom	32	
Output mode	residual on the MINAO Fock anchor	
Level of theory		
Functional	PBE	ωB97X
AO basis	def2-SVP	6-31G(d), spherical d
Coulomb treatment	density fitting	
Auxiliary basis	def2-universal-JFIT	def2-universal-JKFIT
Integration grid	PySCF level 3	
Reference SCF tolerance	10^{-9} Hartree (energy), grid response enabled	
Training data		
Conformation source	GAFF2 classical MD, 100 ns	
Thermostat temperature	500 K or 1600 K (rMD17/MD22); 300 K or 1600 K (3BPA)	
Train / validation conformations	95,000 / 5,000	
Reference DFT calculations used	none	
Optimisation		
Batch size	4 conformations	
Gradient steps per fetched batch	10	
Prepared-geometry pool	8 (2× batch size)	
Pack-building worker processes	8	
Total steps	237,500 (one pass over the training set, 10 steps per batch)	
Optimiser	AdamW, weight decay 10^{-6}	
Learning rate	polynomial decay, $10^{-3} \rightarrow 10^{-5}$	
Gradient clipping	global norm 1.0	
Numerical precision		
Network	fp32	
Physics layer (energy, density map, Fock)	fp64	
Inference force path	fp64 (GPU4PySCF)	

Differentiable physics layer. Training requires a differentiable evaluation of Eq. (1) and of Eq. (4) at every step. We implement the Kohn–Sham energy in PyTorch in `float64`, using density fitting for the Coulomb term (Whitten, 1973; Dunlap, 1983; Vahtras et al., 1993), exchange–correlation quadrature on a Becke grid, and, for the range-separated hybrid, exact exchange computed as a

density-fitted contraction over occupied orbitals. At inference, the Fock build and the analytic gradient are computed with GPU4PySCF (Wu et al., 2025), so forces are obtained from standard quantum-chemistry code evaluated at the predicted density. Integrals, integration grids and the converged SCF calculations used as references follow PySCF (Sun et al., 2020).

B Training Pipeline and Efficient Inference

Overlapping integral construction with optimisation. Section 4.4 states which quantities are prepared ahead of a gradient step; this paragraph gives the arrangement. The worker processes that build them stage their output onto the accelerator on a separate stream, so both preparation and transfer overlap the gradient computation; the anchor Fock matrix, which dominates that cost, is obtained as $\partial E_{\mathrm{KS}}/\partial P$ at the minimal-basis density rather than built separately. Since preparation is the bottleneck, each prepared geometry is used for several steps: we keep a pool larger than the batch size, draw each minibatch from it without replacement and periodically replace the oldest entries, so that a geometry contributes to several steps in a different combination each time and geometries are consumed at the rate they can be prepared (Appendix A).

Efficient inference. Section 4.4 states which properties of the inference procedure make these optimisations available; this paragraph gives their form. Integral screening and setup data are constructed once per geometry and shared by all routines, and the exchange–correlation density and potential from the Fock build are reused in the gradient. Because Eq. (5) and $W = \frac{1}{2}PFP$ need the Fock matrix only in its occupied–occupied block, we build the exchange–correlation contribution in that block alone, which an SCF procedure cannot do because it must update every orbital. Finally, the exchange–correlation quadratures, integration weights, AO screening and the eigendecomposition of Eq. (4) run in single precision with two double-precision refinement sweeps, a computation being kept in double precision whenever the largest quantity it accumulates, times the relative round-off of single precision, would exceed the tolerance required of its output; this criterion retains double precision for the integral derivatives, the core-Hamiltonian terms and the density-fitting metric. The gain from single precision depends on the hardware and is largest on GPUs whose double-precision throughput is far below their single-precision throughput, whereas the structural optimisations are hardware-independent.

C Accuracy Against Converged SCF References

Tables A2 and A3 report errors against our own converged SCF, which isolate the error of the model from differences between programs. Averaged over the ten rMD17 molecules, the force MAE is 0.98 meV/Å and the energy MAE is 0.063 meV for the 1600 K models.

Table A2: **rMD17 against our own converged SCF.** Energy and force MAE at PBE/def2-SVP on the official test frames, for the models trained on 1600 K geometries.

Molecule	Energy MAE (meV)	Force MAE (meV/Å)
Aspirin	0.15	1.78
Azobenzene	0.08	0.94
Benzene	0.002	0.25
Ethanol	0.01	0.51
Malonaldehyde	0.03	1.12
Naphthalene	0.01	0.36
Paracetamol	0.10	1.28
Salicylic acid	0.17	1.69
Toluene	0.01	0.39
Uracil	0.07	1.51

Effect of the sampling temperature. Tables 1 and 3 score both sampling temperatures against the benchmark labels; against our own converged SCF the comparison is the following. On rMD17, raising the sampling temperature from 500 to 1600 K improves the force error on 9 of the 10

Table A3: **3BPA against our own converged SCF.** Energy and force MAE at ωB97X/6-31G(d), for the model trained on 1600 K geometries. The dihedral row pools the three dihedral scans.

Test set	Energy MAE (meV)	Force MAE (meV/Å)
300 K	0.13	1.09
600 K	0.18	1.28
1200 K	0.32	1.76
Dihedral	0.12	1.10

molecules and reduces the mean force MAE from 1.82 to 0.98 meV/Å and the mean energy MAE from 0.375 to 0.063 meV. The largest change is for malonaldehyde ($4.95 \rightarrow 1.12$ meV/Å), where the GAFF2 torsion parameters restrict the 500 K sampling and leave 34% of the test frames outside the training distribution. Molecules whose test frames were already covered at 500 K, such as ethanol, uracil, toluene and naphthalene, also improve, by 1.4–2.5×. Only benzene becomes worse; it is rigid, so hotter sampling adds no conformations it needs, and it already has the lowest error. The 500 K models also have a heavy error tail on the test geometries: their force RMSE against the labels reaches 15.0 meV/Å on uracil and 7.6–8.9 meV/Å on aspirin, salicylic acid and malonaldehyde, against 1.6–3.8 meV/Å for the 1600 K models. On 3BPA, the force MAE of the 1600 K model grows by a factor of 1.6 from the 300 K to the 1200 K test set, and that of the 300 K model by a factor of 7.5 ($1.23 \rightarrow 9.28$ meV/Å); in RMSE, the metric used by the 3BPA baselines, the same comparison gives factors of 1.7 and 10.7 ($1.92 \rightarrow 20.57$ meV/Å). For comparison, the supervised baselines degrade by factors of 6.9–7.4 against the labels. The 1600 K model has no error tail: its ratio of RMSE to MAE matches that of the label noise itself on every test set. The 300 K model at 1200 K does, with its three worst frames at 128 to 167 meV/Å against a median of 11.5. Assigning every test frame to a 15° cell of the benchmark's torsion space shows two different behaviours. On the dihedral scans, where the torsions are essentially the only coordinates that vary, the control's error is concentrated in cells its trajectory never visited (1.4–2.3 meV/Å against 0.5–0.6 in well-covered cells). At 1200 K it is not: the error is 9.5 meV/Å in never-visited cells and 9.1 in well-covered ones, so what a 300 K trajectory lacks at that temperature is not a torsion angle.

D Molecular Dynamics of Ac-Ala$_3$-NHMe

Low errors on independent frames do not by themselves guarantee stable dynamics. We therefore run molecular dynamics driven directly by the model, with one forward pass, one Fock build and one gradient evaluation per step and no SCF iteration, and compare against Born–Oppenheimer MD at the same level of theory. Both trajectories start from identical positions and velocities and use the same integrator, timestep (1 fs) and random stream, so the comparison is paired. We run 100 ps with a Langevin thermostat in each arm, together with a 10 ps microcanonical pair for the energy-drift measurement.

Table A4: **Model-driven molecular dynamics of Ac-Ala$_3$-NHMe** compared with Born–Oppenheimer MD at the same level of theory (PBE/def2-SVP), from identical initial conditions. Drift is measured on the 10 ps microcanonical runs, as a 95% interval over the four block slopes of that window after a 0.5 ps burn-in; neither drift is resolved from zero. The temperature is measured on the 100 ps thermostatted runs.

Driver	Energy drift (meV/atom/ps)	Mean temperature (K)	Cost per step (s)
Converged DFT	$[-0.071, +0.121]$	283.1	3.72
Variational (no SCF)	$[-0.078, +0.192]$	284.1	**1.09**

Table A4 summarises the outcome. The model-driven trajectory is stable for the whole 100 ps: no bond breaks, the molecule stays connected on every frame, and no step produces an invalid density. Neither microcanonical drift is resolved from zero, and nor is the excess between them, whose 95% interval is $[-0.198, +0.262]$ meV/atom/ps; pooling two further starting states bounds that excess at 0.09 meV/atom/ps (one-sided 95%). In the thermostatted runs no difference between

the trajectories is resolved in bond and angle distributions, sampled conformational observables, the vibrational spectrum or the mean temperature. They separate to 0.5 Å heavy-atom RMSD within a few picoseconds, but so does a DFT trajectory started from a 10^{-3} Å perturbation of the same initial condition, so the separation reflects the chaotic dynamics rather than the model. At less than a third of the cost per step, 100 ps takes 30 GPU-hours instead of 104.

E One Model Trained Across Eight Ligands

Table A5: **One model trained across eight ligands** of the galectin-8 target of a protein–ligand benchmark set (Hahn et al., 2022), at ωB97X/6-31G(d) with an added sulfur basis. Errors are against our own converged SCF on 100 held-out frames per ligand. Unlike the published DFT-MD frames of rMD17, these are the tail of the same classical trajectories the training geometries come from, separated by a 0.5 ns gap. The centred row subtracts the mean signed error over all frames, the residual that survives in an energy difference.

Ligand	ligF	ligOMe	ligNHMe	ligNMe2	ligOEt	ligPyr	ligOH	ligNH2
Number of atoms	52	56	57	60	59	64	53	54
Energy MAE (meV)	0.31	0.32	0.34	0.37	0.36	0.41	0.31	0.32
Centred energy MAE (meV)	0.06	0.06	0.06	0.08	0.07	0.07	0.06	0.07
Force MAE (meV/Å)	0.94	0.91	0.95	0.94	0.93	0.96	0.90	0.93

The experiments of Section 5 use one model per molecule, the standard setting on rMD17. Since Eq. (3) is defined for any closed-shell system in a given basis, one model can also be trained on several molecules. We test this on eight ligands of the galectin-8 target of a protein–ligand free-energy benchmark set (Hahn et al., 2022), considerably larger than the molecules of the main text at 52 to 64 atoms against 21 for aspirin, training a single model on classical-MD geometries of all eight at ωB97X/6-31G(d).

Table A5 reports the result. The force error is close to that of the per-molecule models on the smaller rMD17 systems, every frame yields an admissible density, and no ligand is carried by the others: the errors span only 6% across the eight, and although the largest ligand has the highest error, the margin is too narrow for any appreciable size dependence. The energy error is dominated by a constant shift, with all 800 frames erring in the same direction and the shift carrying 90 to 95% of the mean-square error. Free-energy applications consume energy differences, in which such a shift cancels, so the relevant residual is the centred row, smaller than the raw error by a factor of 4.8 to 5.8. Starting an SCF calculation from the predicted density saves 54% of its iterations (Appendix I), as for the single-molecule models.

F Supervised Training of the Same Network on 3BPA

Section 5.3 compares variational with supervised training on four rMD17 molecules at PBE/def2-SVP. Table A6 repeats that comparison at ωB97X/6-31G(d) on 3BPA, where the test sets leave the training distribution. Both arms use the same $95{,}000$ training geometries, network, optimiser, schedule, seed and GPU type, and the same evaluation code on the same complete test sets; they differ only in the training signal. Errors are against our own converged SCF, which removes the benchmark labels from the comparison and leaves the two training signals as the only difference.

The force error of the supervised model is twice that of the variational model, and the ratio is between 1.9 and 2.1 in MAE on every one of the six test sets. The gap is also the same inside every torsion-coverage group: 2.00 in cells the shared training trajectory visited often and 2.05 in cells it visited rarely. Since both models were trained on the same geometries, a gap that does not depend on where the test frames lie reflects the training signal rather than the sampling. The energy gap is larger, a factor of 31 pooled over the test sets, and it has a structural cause: the variational objective *is* the Kohn–Sham energy of the predicted density, so the model's energy agrees with its own converged SCF to a few tenths of a meV, whereas the supervised loss fits Fock-matrix elements and never evaluates an energy. As an initial guess for a subsequent SCF calculation the variational model is

Table A6: **Variational versus supervised training of the same network on 3BPA.** Energy and force MAE against our own converged SCF at ωB97X/6-31G(d), on the benchmark's six test sets, with the three dihedral scans pooled. Both models are trained on the same geometries sampled at 1600 K.

	Energy MAE (meV)		Force MAE (meV/Å)	
Test set	supervised	variational	supervised	variational
300 K	4.35	**0.13**	2.18	**1.09**
600 K	5.51	**0.18**	2.66	**1.28**
1200 K	8.32	**0.32**	3.69	**1.76**
Dihedral	4.07	**0.12**	2.15	**1.10**

also the better of the two here, saving 6.60 of 12.49 iterations against 5.89 for the supervised model, in contrast to rMD17, where the two were equivalent.

The two routes cost 32.3 and 31.5 GPU-hours respectively, a difference within node-to-node variation. 3BPA is a 27-atom molecule in a small basis, so its labels are cheap, and the cost argument of Section 5.3 does not carry the result here; the accuracy difference stands on its own.

G The Degeneracy-Safe Backward Pass

The generic backward pass of a symmetric eigendecomposition differentiates every eigenvector individually and so contains factors $1/(\varepsilon_i - \varepsilon_j)$ for all pairs of eigenvalues, including pairs within the occupied block. Molecular core levels are often nearly degenerate, so these factors overflow and training diverges within a few steps, even though their contributions to the gradient of a subspace-invariant loss cancel exactly. The resolution is that the objective never needs the eigenvectors individually, only the occupied projector $\Pi = U_{:,1:N_{\mathrm{occ}}} U_{:,1:N_{\mathrm{occ}}}^{\top}$, which is invariant under any rotation within the occupied block and within the virtual block. The derivative of an invariant quantity cannot depend on the gaps it is invariant to, so those denominators must cancel; carrying that cancellation out analytically rather than numerically gives a backward pass for the map $\tilde{M} = X^{\top} M_\theta X \mapsto \Pi$ in which only occupied–virtual differences appear:

$$\frac{\partial \Pi}{\partial \tilde{M}}\left[\bar{\Pi}\right] = UKU^{\top}, \qquad K_{ia} = \frac{\left(U^{\top}\bar{\Pi}\,U\right)_{ia}}{\varepsilon_i - \varepsilon_a}, \quad i \le N_{\mathrm{occ}} < a, \tag{6}$$

where the occupied–occupied and virtual–virtual entries of K are set to zero, so the smallest denominator is the HOMO–LUMO gap rather than the spacing between core levels and training remains stable throughout. Two implementation notes. The upstream gradient must be symmetrised before the contraction, and the occupied–virtual block must be written into both triangles with a factor $\frac{1}{2}$, since otherwise every contraction counts that block twice; the symptom of getting this wrong is an analytic gradient exactly twice the finite-difference slope. We gate the density map against finite differences before every training run.

H The Force Readout

Table A7 compares four ways of turning the same predicted density into forces. All four are computed in a single evaluation pass per frame, from one predicted density and against one converged SCF calculation performed in the same pass, so they differ only in the readout.

Eigenvalues of the predicted matrix. Reading forces from the eigenvalues of M_θ directly, as is common for supervised Hamiltonian predictors, gives errors of 334 to 526 meV/Å, comparable in magnitude to the forces themselves and about $500\times$ the error of the recipe we use. This is the expected consequence of Section 4.2: the objective does not constrain those eigenvalues. Rebuilding the Fock matrix from the predicted density is what removes that error, and the remaining choices concern how the rebuilt matrix is used.

Occupied-block canonicalisation. Section 4.3 rotates the model's occupied orbitals to diagonalise $F_{oo} = C_{\mathrm{occ}}^{\top} F[P]\, C_{\mathrm{occ}}$ before forming the energy-weighted density W. The alternative — taking the

diagonal of F_{oo} as orbital energies, i.e. Rayleigh quotients in the model's own orbital basis — discards the occupied–occupied off-diagonal block, and the Pulay term of Eq. (2) is then wrong by exactly that block. The model's orbitals are never canonical for $F[P]$, so the block is never zero, and the resulting error is two orders of magnitude larger than ours. At a converged reference density, where the two recipes must agree, the canonicalised form reproduces the exact analytic gradient to 10^{-9} meV/Å while the uncanonicalised form is off by four orders of magnitude — which is how the defect was found. The size of the gap is specific to the zero-step recipe: the same Rayleigh-quotient readout taken after a full Roothaan step, where the orbitals do diagonalise the Fock matrix of the preceding density, is only about $4\times$ worse than ours. What the table supports is therefore that canonicalisation is what makes the zero-step recipe work, not that it is worth two orders of magnitude in general.

Why not take one Roothaan step first. A natural-seeming refinement is to perform one full SCF update from the predicted density — diagonalise $F[P_\theta]$, rebuild the density, then differentiate — on the grounds that this moves the density towards the fixed point. It does not. The undamped SCF map has a charge-sloshing eigenvalue of magnitude 2–3 for several of the molecules studied, so a single bare step overshoots; the same behaviour is visible in a conventional solver, whose first two DIIS cycles also move away before extrapolation takes hold. Empirically the one-step variant is worse on nine of the ten molecules, by $1.8\times$ in geometric mean, and on ethanol the two agree to within one standard error of the test-set mean. We therefore rebuild the Fock matrix but never re-diagonalise it outside the N_{occ}-dimensional occupied block.

Table A7: **Ablation of the force readout**, force MAE (meV/Å) against our own converged SCF at PBE/def2-SVP, on the complete official rMD17 test split and on the same checkpoints as Table A2. Canonicalised W is the readout of Section 4.3, used everywhere else in this paper. The last row is the geometric mean over molecules of each readout's ratio to it. The best readout for each molecule is in bold.

Molecule	Plug-in eigenvalues	Rayleigh-quotient W	One Roothaan step	Canonicalised W
Aspirin	463	126	3.69	**1.78**
Azobenzene	342	96.8	1.84	**0.94**
Benzene	349	81.8	0.43	**0.25**
Ethanol	458	90.8	**0.51**	**0.51**
Malonaldehyde	489	113	2.28	**1.12**
Naphthalene	334	83.5	0.63	**0.36**
Paracetamol	459	127	2.51	**1.28**
Salicylic acid	397	114	3.97	**1.69**
Toluene	395	70.8	0.57	**0.39**
Uracil	526	204	3.24	**1.51**
Geometric-mean ratio	$518\times$	$132\times$	$1.8\times$	—

Invariance to the unconstrained blocks. The claim in Section 4.3 that the reported force depends on M_θ only through its occupied eigenspace is verified numerically: perturbing the occupied–occupied or virtual–virtual blocks of M_θ changes the canonicalised force by at most 10^{-8} meV/Å, while occupied–virtual perturbations of the same magnitude move it at the expected scale.

I Initialising Conventional SCF

Because the predicted density is admissible, it can also serve as the initial guess of an unmodified SCF solver when a fully converged result is required. Tables A8 and A9 show that this halves the number of SCF iterations relative to the standard minimal-basis guess, consistently across a generalised-gradient and a range-separated hybrid functional. Averaged over the ten rMD17 molecules, the count falls from 12.67 to 6.07 iterations for the 1600 K models (-52%) and to 6.23 for the 500 K models (-51%); on the six 3BPA test sets it falls from 12.50 to 5.89 (-53%). The multi-ligand model of Appendix E behaves the same way, saving 7.01 of 13.06 iterations (-54%) pooled over its eight ligands. Every test frame required fewer iterations and none failed to converge. This comparison is relevant for unordered geometries, where no previous density is available; in sequential molecular dynamics, extrapolation from previous frames recovers much of the advantage, which is why our

timing comparisons use DFT warm-started from the preceding frame. On rMD17 the MINAO column is shared between the two sampling temperatures because the test frames and the reference solver are identical.

Table A8: **SCF iterations on rMD17**, mean per molecule over the complete official test split, at PBE/def2-SVP. Standard errors of the saved iterations are at most 0.03. 100% of frames improved in every row; no frame failed to converge.

		Model		Saved	
Molecule	MINAO	500 K	1600 K	500 K	1600 K
Aspirin	14.91	7.17	7.04	7.74	7.87
Azobenzene	13.01	7.12	6.97	5.89	6.04
Benzene	10.26	4.23	4.93	6.03	5.33
Ethanol	9.96	4.93	4.76	5.04	5.20
Malonaldehyde	11.86	6.84	5.91	5.02	5.95
Naphthalene	11.82	5.77	5.47	6.04	6.35
Paracetamol	14.87	6.93	6.90	7.94	7.97
Salicylic acid	14.12	6.76	6.93	7.36	7.19
Toluene	11.01	5.43	5.13	5.58	5.88
Uracil	14.85	7.11	6.62	7.74	8.24

Table A9: **SCF iterations on 3BPA**, mean over each complete test set at ωB97X/6-31G(d), for the model trained on 1600 K geometries. Intervals are standard errors of the paired per-frame difference; the dihedral row pools the three scans. No frame failed to converge.

Test set	MINAO	Model	Saved
300 K	12.63	5.85	6.78 ± 0.01
600 K	12.74	6.02	6.71 ± 0.01
1200 K	12.86	6.38	6.48 ± 0.01
Dihedral	12.26	5.70	6.56 ± 0.01