Graph Neural Network Interatomic Potential Ensembles with Calibrated Aleatoric and Epistemic Uncertainty on Energy and Forces[†]

Jonas Busk[a,*], Mikkel N. Schmidt[b], Ole Winther[b,c,d], Tejs Vegge[a] and Peter Bjørn Jørgensen[a]

Inexpensive machine learning potentials are increasingly being used to speed up structural optimization and molecular dynamics simulations of materials by iteratively predicting and applying interatomic forces. In these settings, it is crucial to detect when predictions are unreliable to avoid wrong or misleading results. Here, we present a complete framework for training and recalibrating graph neural network ensemble models to produce accurate predictions of energy and forces with calibrated uncertainty estimates. The proposed method considers both epistemic and aleatoric uncertainty and the total uncertainties are recalibrated post hoc using a nonlinear scaling function to achieve good calibration on previously unseen data, without loss of predictive accuracy. The method is demonstrated and evaluated on two challenging, publicly available datasets, ANI-1x (Smith *et al.*[1]) and Transition1x (Schreiner *et al.*[2]), both containing diverse conformations far from equilibrium. A detailed analysis of the predictive performance and uncertainty calibration is provided. In all experiments, the proposed method achieved low prediction error and good uncertainty calibration, with predicted uncertainty correlating with expected error, on energy and forces. To the best of our knowledge, the method presented in this paper is the first to consider a complete framework for obtaining calibrated epistemic and aleatoric uncertainty predictions on both energy and forces in ML potentials.

1 Introduction

Accurate and computationally inexpensive machine learning (ML) potentials are increasingly being used to accelerate atomic structure optimization and molecular dynamics simulations by iteratively predicting and applying interatomic energies and forces[3,4]. This development has the potential to revolutionise disciplines of computational chemistry such as predicting molecular properties and structures, predicting reaction mechanisms and networks, as well as discovering new materials, e.g., for energy conversion and storage of renewable energy. In these settings, it is crucial to asses the confidence of predictions and to detect when predictions are unreliable, to avoid wrong or misleading results by either ending the simulation early or enabling recovery by falling back to higher fidelity but also more computationally expensive methods, such as density functional theory (DFT)[5]. Uncertainty quantification (UQ) methods can enable assessment of the confidence in predictions and thus make applications of ML potentials more robust and reliable. To ensure uncertainty estimates are useful and informative, they need to be calibrated, i.e. there should be an agreement between the predictive distribution and the empirical distribution. Especially if the predicted uncertainty is expected to indicate the range of plausible values. For example in a screening application where candidate materials are filtered for specific useful properties, instances with poor point estimates but with high uncertainty could still potentially be interesting and should not be discarded. Good calibration thus ensures that ML-based uncertainty estimates are interpretable and actionable, and enable the selection of a suitable confidence threshold for a given application on the original unit scale of the quantity of interest.

When considering predictive uncertainty it is often useful to distinguish between *epistemic uncertainty* and *aleatoric uncertainty*[6,7]. Epistemic uncertainty arises from uncertainty in determining the model parameters and can in principle be reduced by observing more data. On the other hand, aleatoric uncertainty can originate from random noise or inconsistency in the data, or from inadequacy of the model to fit the data precisely, and can therefore generally not be reduced by observing more data. A widely used method for estimating epistemic uncertainty in ML

[a] *Department of Energy Conversion and Storage, Technical University of Denmark, Kongens Lyngby, Denmark. E-mail: {jbusk,teve,pbjo}@dtu.dk.*

[b] *Department of Applied Mathematics and Computer Science, Technical University of Denmark, Kongens Lyngby, Denmark. E-mail: {mnsc,olwi}@dtu.dk.*

[c] *Center for Genomic Medicine, Rigshospitalet, Copenhagen University Hospital, Denmark.*

[d] *Bioinformatics Centre, Department of Biology, University of Copenhagen, Denmark.*

[*] *Corresponding author: jbusk@dtu.dk.*

[†] Electronic Supplementary Information (ESI) available.

potentials is to apply an ensemble of models and use the agreement of the predictions of the ensemble members as a measure of confidence in the prediction[8]. This approach relies on the observation that randomly initialised models will often provide increasingly different predictions further away from the training data distribution. From a Bayesian perspective, if the individual ensemble members are seen as draws from the posterior distribution, the variance between predictions of the ensemble members are a measure of the posterior uncertainty[9–11]. Other popular approaches for estimating epistemic uncertainty with neural networks include Bayesian neural networks[12,13] that can directly learn a probability distribution over the neural network parameters, and Monte Carlo dropout[14] that estimates the predictive distribution through multiple stochastic forward passes. However, UQ methods that only account for epistemic uncertainty and thus ignore aleatoric uncertainty are not inherently calibrated, which makes it difficult to select an appropriate confidence threshold for a given application. Methods for estimating the aleatoric uncertainty include the mean-variance model[15], that explicitly predicts the uncertainty variance as an additional model output, and more recently evidential learning[16,17] that learns the parameters of a higher order distribution over the likelihood parameters, and conformal prediction[18], a distribution-free approach that estimates a prediction interval directly. The deep ensemble approach[19] combines an ensemble of mean-variance neural network models to estimate both aleatoric and epistemic uncertainty in a unified model.

As discussed above, an important aspect of predictive uncertainty is the concept of *calibration*, which implies an agreement between the predicted uncertainty and the expected empirical error. When evaluating calibration, it is important to consider the asymmetric relationship between errors and uncertainties. By the common assumption that errors are drawn from a distribution (usually Gaussian), small uncertainties should be associated with small errors and large errors should be associated with large uncertainties, but large uncertainties can be associated with both small and large errors. Therefore there is no direct correlation between uncertainties and the magnitude of errors. However, there should be a correlation between the uncertainties and the expected magnitude of errors. Several works have proposed methods for evaluating and validation calibration of regression models. Kuleshov *et al.*[20] proposed evaluating the coverage of the errors by the predictive distribution averaged over the data using a calibration curve. Later, Levi *et al.*[21] proposed checking the correlation of uncertainties and expected errors computed in bins of increasing uncertainty in a reliability diagram. Recently, Pernot[22] highlighted the limitations of the previous approaches and proposed an additional analysis of z-scores (standard scores) for variance-based UQ methods. UQ and calibration for molecular property prediction and interatomic ML potentials has been explored in recent literature[18,23–26], but often the training and inference methods used do not inherently ensure good calibration. The method presented in this paper is, to the best of our knowledge, the first to consider a complete framework for obtaining calibrated epistemic and aleatoric uncertainty predictions on both energy and forces.

In previous work, we have shown how to extend a graph neural network model for predicting formation energy of molecules to also provide calibrated uncertainty estimates that can be decomposed into aleatoric and epistemic uncertainty[27]. The method works by combining an ensemble of models with mean-variance outputs and applying post hoc recalibration with isotonic regression on data not used for the training. In this work, we further extend the approach to include calibrated uncertainty on the force predictions. Specifically, we extend a neural network potential with a probabilistic predictive distribution on energy and forces, and consider a deep ensemble of models[19] to express the aleatoric and epistemic uncertainty about the energy and force predictions. The uncalibrated predictive distributions are then recalibrated post hoc to fit the error distribution on previously unseen data. An added benefit of this approach is that ensemble models are generally known to produce more accurate predictions than single models[8]. Through computer experiments, we demonstrate that the proposed method results in accurate and calibrated predictions on two publicly available datasets, ANI-1x[1] and Transitions1x[2] containing out-of-equilibrium and near-transition-state structures, respectively. The main contribution of the work is a complete framework for training and evaluating neural network potentials with accurate predictions and calibrated aleatoric and epistemic uncertainty on both energies and forces.

The rest of the paper is structured as follows. The proposed method including the extended graph neural network model and the recalibration procedure is described in Section 2. The datasets, experiments and results are presented in Section 3. Finally, the main findings and perspectives are discussed in Section 4 and we conclude in Section 5.

2 Methods

2.1 Graph neural network model

As the base model for our ensemble we use PaiNN[28], an equivariant message passing neural network (MPNN) model designed specifically for predicting properties of molecules and materials. The model provides a mapping from sets of atomic species and positions $\{(Z_i, \vec{r}_i)\}$ to potential energy E and interatomic forces $\{\vec{F}_i\}$. The potential energy is modelled as a sum over the atomic contributions E_i:

$$E = \sum_i E_i, \tag{1}$$

and the forces are computed as the derivative of the potential energy with respect to the atomic positions, ensuring conservation of energy:

$$\vec{F}_i = -\partial E / \partial \vec{r}_i. \tag{2}$$

Specifically, the model input is represented as a graph, where there is an edge between a pair of atoms if the mutual distance between the atoms is below a certain cutoff. The cutoff distance is treated as a hyperparameter and is fixed at 5.0 Å in all of our experiments. The neural network architecture consists of a number of interaction layers, where information, or "messages", are exchanged along the edges of the input graph to update the hidden node states, followed by a readout function represented by a

fully connected neural network that outputs the atom-wise quantities. The number of interaction layers and the size of the node hidden states are hyperparameters of the model.

2.2 Extended model with aleatoric uncertainty

We extend the base model with additional outputs representing the aleatoric energy uncertainty $\sigma_E^2 = \sum_i \sigma_{E_i}^2$ and atom-wise aleatoric force uncertainties $\{\sigma_{F_i}^2\}$. The atom-wise quantities, $\sigma_{E_i}^2$ and $\sigma_{F_i}^2$, are constrained to be positive by passing them through a softplus activation function, $\log(1 + \exp(\cdot))$, and adding a small constant for numerical stability. Note that here we chose to represent the atom-wise aleatoric force uncertainty by a single scalar even though the force vectors are 3-dimensional. This simplifying assumption means that we consider the noise scale in the spatial dimensions to be isotropic, i.e., uniform in all directions. Other options would be to represent the aleatoric force uncertainty by a common scalar for all atoms or as atom-wise vectors representing the uncertainty in each direction. However we found the isotropic approach to be a reasonable compromise and to work well in practice and did not study the other solutions further.

2.3 Model training procedure

Each network in the ensemble is initialized with random weight parameters θ and trained individually on the same training dataset using a loss function composed of a weighted sum of the energy and force loss terms:

$$\mathscr{L}(\theta) = \lambda_E \mathscr{L}_E(\theta) + \lambda_F \mathscr{L}_F(\theta), \tag{3}$$

where the weight λ_F is between 0 and 1 and $\lambda_E = (1 - \lambda_F)$.

ML potentials are usually trained with mean squared error (MSE) loss for both the energy and forces. Using a negative log likelihood (NLL) loss function provides a natural way of training mean-variance models that also consider uncertainty[15]. The mean squared error (MSE) loss for the energy is straight forward. The MSE loss for the forces is evaluated per atom and component-wise over the spatial dimensions and is then averaged over the number of atoms. The negative log likelihood (NLL) loss for energy, assuming a normally distributed error, is given for a single instance by the following expression where $x = \{(Z_i, \vec{r}_i)\}$ represents the model input and the observed values of energy and forces are denoted by E^{obs} and F^{obs}, respectively:

$$\text{NLL}_E(\theta) = -\log p(E^{\text{obs}}|x, \theta) \tag{4}$$

$$= \frac{1}{2}\left(\frac{\left(E^{\text{obs}} - E(x)\right)^2}{\sigma_E^2(x)} + \log \sigma_E^2(x) + \log 2\pi \right). \tag{5}$$

The instance-wise energy losses are then averaged over the number of instances.

Analogous to the MSE loss for forces, the NLL loss for forces is evaluated per atom i and component-wise over the spatial dimensions D (recall that the predicted atom-wise force uncertainty $\sigma_{F_i}^2$

is a single scalar applied over all spatial dimensions):

$$\text{NLL}_{F_i}(\theta) = \sum_{d=1}^{D} -\log p(F_{i,d}^{\text{obs}}|x, \theta) \tag{6}$$

$$= \sum_{d=1}^{D} \frac{1}{2}\left(\frac{\left(F_{i,d}^{\text{obs}} - F_{i,d}(x)\right)^2}{\sigma_{F_i}^2(x)} + \log \sigma_{F_i}^2(x) + \log 2\pi \right). \tag{7}$$

The atom-wise force losses are then averaged over the total number of atoms. Note that NLL with fixed variance is equivalent to (scaled) MSE and the $\log 2\pi$ terms are constant and can be omitted in training. Here, for models that are trained with a combination of MSE and NLL loss on either energy or forces, we scale the NLL loss by the expected uncertainty (determined empirically) to avoid the NLL loss dominating.

Training directly with NLL loss can be unstable due to interactions between the mean and variance in the loss function, so we apply a training procedure similar to previous work[27], where the model is always trained with MSE loss for an initial warmup period before linearly interpolating to the NLL loss. Other more sophisticated methods for training with NLL loss exist[29,30], but we found this simple approach to be sufficient to achieve training stability in our experiments.

2.4 Ensemble model with epistemic uncertainty

To estimate the epistemic uncertainty, we follow the approach of Lakshminarayanan et al.[19] and make an ensemble approximation by combining the predictions of M individual models. Using a Bayesian interpretation of deep ensemble models[9–11], we can interpret the model weights $\theta^{(m)}$ of each ensemble member m as samples from an approximate posterior distribution $q(\theta) \approx p(\theta|\mathscr{D})$, where $\mathscr{D}$ is the training data. For a regression model with input x and output y trained on a dataset $\mathscr{D}$ we have:

$$p(y|x, \theta) = \int p(y|x, \theta)p(\theta|\mathscr{D})d\theta, \tag{8}$$

$$\approx \frac{1}{M}\sum_{m=1}^{M} p(y|x, \theta^{(m)}), \quad \theta^{(m)} \sim p(\theta|\mathscr{D}), \tag{9}$$

$$\approx \frac{1}{M}\sum_{m=1}^{M} p(y|x, \theta^{(m)}), \quad \theta^{(m)} \sim q(\theta). \tag{10}$$

The first approximation is to estimate the integral with M samples from the distribution $p(\theta|\mathscr{D})$ and the second approximation comes from approximating the true posterior $p(\theta|\mathscr{D})$ with the distribution $q(\theta)$. The uncertainty arising from $p(y|x, \theta)$ is the aleatoric uncertainty, while the epistemic uncertainty is modeled as the uncertainty arising from the distribution of the model parameters $q(\theta)$.

When applying this interpretation to a ML potential energy model, the underlying assumption of the model is that energy and force observations are generated by first sampling θ and then sampling the normally distributed noise, i.e., for a single molecu-

lar energy and atomic force we then have:

$$\theta \sim q(\theta), \tag{11}$$

$$E^{\mathrm{obs}}(x) \sim \mathcal{N}\left(E_\theta(x), \sigma^2_{\theta,E}(x)\right), \tag{12}$$

$$\vec{F}_i^{\mathrm{obs}}(x) \sim \mathcal{N}\left(-\frac{\partial E_\theta}{\partial \vec{r}_i}(x), \mathbf{I}\sigma^2_{\theta,F_i}(x)\right). \tag{13}$$

We have two levels of stochastic variables, so to calculate the variance we can use the law of total variance:

$$\mathrm{Var}\,[Y] = \underbrace{\mathbb{E}\left[\mathrm{Var}\,[Y|X]\right]}_{\text{aleatoric}} + \underbrace{\mathrm{Var}\left[\mathbb{E}\,[Y|X]\right]}_{\text{epistemic}}. \tag{14}$$

Using the law of total variance, we get the following expression for the observed energy variance:

$$\mathrm{Var}\left[E^{\mathrm{obs}}\right] = \mathbb{E}\left[\mathrm{Var}\left[E^{\mathrm{obs}}|\theta\right]\right] + \mathrm{Var}\left[\mathbb{E}\left[E^{\mathrm{obs}}|\theta\right]\right], \tag{15}$$

$$= \underbrace{\mathbb{E}_\theta\left[\sigma^2_{\theta,E}(x)\right]}_{\text{aleatoric}} + \underbrace{\mathrm{Var}_\theta\left(E_\theta(x)\right)}_{\text{epistemic}}. \tag{16}$$

Since the force observation is a vector, we compute the total variance element-wise:

$$\mathrm{Var}\left[F_{i,d}^{\mathrm{obs}}, F_{i,d}^{\mathrm{obs}}\right] = \mathbb{E}\left[\mathrm{Var}\left[F_{i,d}^{\mathrm{obs}}, F_{i,d}^{\mathrm{obs}}|\theta\right]\right] \tag{17}$$

$$+ \mathrm{Var}\left[\mathbb{E}\left[F_{i,d}^{\mathrm{obs}}|\theta\right], \mathbb{E}\left[F_{i,d}^{\mathrm{obs}}|\theta\right]\right]$$

$$= \underbrace{\mathbb{E}_\theta\left[\sigma^2_{\theta,F_i}(x)\right]}_{\text{aleatoric}} + \underbrace{\mathrm{Var}_\theta\left(-\frac{\partial E_\theta(x)}{\partial r_{i,d}}, -\frac{\partial E_\theta(x)}{\partial r_{i,d}}\right)}_{\text{epistemic}}. $$
$$\tag{18}$$

Treating the parameters of the ensemble member as samples from an approximate posterior $q(\theta)$ and using the samples to approximate the expectations, we get the following expressions for the energy mean and variance:

$$E^{(*)} = \frac{1}{M}\sum_{m=1}^M E^{(m)}, \tag{19}$$

$$\sigma^2_{E^{(*)}} = \underbrace{\frac{1}{M}\sum_{m=1}^M \sigma^2_{E^{(m)}}}_{\text{aleatoric}} + \underbrace{\frac{1}{M}\sum_{m=1}^M\left(E^{(m)} - E^{(*)}\right)^2}_{\text{epistemic}}. \tag{20}$$

Similarly, the mean and variance of the forces for a single atom i are given by the following expressions:

$$\vec{F}_i^{(*)} = \frac{1}{M}\sum_{m=1}^M \vec{F}_i^{(m)}, \tag{21}$$

$$\sigma^2_{F_i^{(*)}} = \underbrace{\frac{1}{M}\sum_{m=1}^M \sigma^2_{F_i^{(m)}}}_{\text{aleatoric}} + \underbrace{\frac{1}{M}\sum_{m=1}^M \frac{1}{D}\left\|\vec{F}_i^{(m)} - \vec{F}_i^{(*)}\right\|^2}_{\text{epistemic}}, \tag{22}$$

where D denotes the spatial dimensions. With the assumption of isotropic force variance, we average across the spatial dimension in (22) when estimating the epistemic force variance of (18) using the sample variance. The mean energy and forces represent the ensemble prediction and the energy and force variances represent the ensemble total uncertainties, which can be decomposed into aleatoric and epistemic components as shown above.

When we want to evaluate the likelihood of an observation, we need access to the predictive distribution and knowing the mean and variance is not sufficient. Following Lakshminarayanan *et al.*[19], we parameterize a normal distribution with the mean and variance. With the mean and variance specified, the normal distribution is the maximum entropy probability distribution, i.e., we make the least assumptions about the data by using a normal distribution following the maximum entropy principle[31,32]. However, to hold true, this would require all the variance outputs of the ensemble members to be equal. If the variances follow an inverse-gamma distribution, the predictive distribution would be a student-t in the infinite ensemble limit, which is the assumption used in deep evidential regression[16,29].

2.5 Uncertainty calibration

Several methods exist for evaluating the calibration of regression models. NLL provides a standard metric for quantifying the overall quality of probabilistic models by measuring the probability of observing the data given the predicted distribution. However, NLL depends on both the predicted mean and uncertainty (see eq. 5 and 7) and it can be useful to evaluate only the quality of the uncertainty estimates. For example, it is often informative to visually compare the predicted uncertainties with the empirical errors by plotting them. Since the total uncertainty of the ensemble model (eq. 20 and eq. 22) can be interpreted as a variance of a normal distribution, we expect most of the errors to lie within 2-3 standard deviations of the predictive distribution. The variance-based approach also allows us to evaluate standard scores, also known as z-scores, defined as the empirical error divided by the standard deviation of the predictive distribution[22]:

$$z = \frac{y^{\mathrm{obs}} - y(x)}{\sigma(x)}. \tag{23}$$

A z-score variance (ZV) close to 1 indicates that the predicted uncertainty on average corresponds to the variance of the error and thereby is an indication of good average calibration. The same approach can be applied to subsets of the data leading to a local z-score variance (LZV) analysis, for example by evaluating ZV in equal size bins of increasing uncertainty to test the consistency of the uncertainty estimates[22]. For plotting, we found it useful to report the square root of the z-variance (RZV). Additionally, we can assess how well the uncertainty estimates correspond to the expected error locally by sorting the predictions in equal size bins by increasing uncertainty and plotting the root mean variance (RMV) of the uncertainty versus the empirical root mean squared error (RMSE), also known as an error-calibration plot or reliability diagram. The error-calibration can be summarized by the expected normalized calibration error (ENCE)[21], which measures the mean difference between RMV and RMSE normalised

by RMV:

$$\text{ENCE} = \frac{1}{K} \sum_{k=1}^{K} \frac{|\text{RMV}_k - \text{RMSE}_k|}{\text{RMV}_k}, \qquad (24)$$

where $k = 1, \dots, K$ iterates the bins. LZV analysis and the reliability diagram provide two useful ways to evaluate the local consistency of the uncertainty estimates. We use 15 equal sized bins in all of our analyses.

Good average or local calibration is not sufficient to ensure that individual uncertainty estimates are informative, i.e., if the uncertainty estimates are homoscedastic, they are not very useful. Thus it is generally desirable for uncertainty estimates to be as small as possible while also having some variation, which is also known as sharpness. To measure sharpness, we report the root mean variance (RMV) of the uncertainty, which should be small and correspond to the RMSE, and the coefficient of variation (CV), which quantifies the ratio of the standard deviation of the uncertainties with the mean uncertainty and thus the overall dispersion, or heteroscedasticity, of the predicted uncertainty:

$$\text{CV} = \frac{\sqrt{N^{-1} \sum_{n=1}^{N} (\sigma(x_n) - \overline{\sigma})^2}}{\overline{\sigma}}, \qquad (25)$$

where $n = 1, \dots, N$ in this case iterates the test dataset, $\sigma(x_n)$ is the predicted standard deviation (uncertainty) of instance n and $\overline{\sigma} = N^{-1} \sum_{n=1}^{N} \sigma(x_n)$ is the mean predicted standard deviation. If uncertainties are heteroscedastic while having good local calibration, it is also an indication of good ranking ability which is important in certain applications such as active learning[8].

2.6 Uncertainty recalibration

The model training procedure described above does not by itself ensure good uncertainty calibration when the model is applied to unseen data. The individual models may overfit to the training data and the total ensemble uncertainties (eq. 20 and 22) are strictly greater than any of the individual model uncertainties, and not fitted on any data. Therefore, following the approach of our previous work, Busk *et al.*[27], we recalibrate the ensemble uncertainty estimates post hoc by using a recalibration function that maps the uncalibrated predictive distribution to a calibrated distribution. The recalibration function is a non-linear uncertainty scaling function based on isotonic regression fitted to predict empirical squared errors on the validation set. Specifically, the recalibration function maps the uncalibrated uncertainty estimates σ^2 to scaled uncertainty estimates $s^2 \sigma^2$, where s^2 is the predicted scaling factor. In our experiments, both energy uncertainties $\sigma^2_{E^{(*)}}$ and force uncertainties $\sigma^2_{F^{(*)}}$ are recalibrated in this way.

Because the recalibration function is a scaling function, the recalibration procedure does not change the mean of the predictive distribution and thus does not change the prediction. Additionally, the isotonic regression results in a monotonic increasing scaling function and thus preserves the ordering of the uncertainty estimates and thereby the ranking. We use the implementation of isotonic regression available in the scikit-learn Python package[33].

3 Experiments and Results

3.1 Datasets

The proposed method was evaluated on two publicly available datasets designed specifically for the development and evaluation of ML potentials, ANI-1x[1] and Transitions1x[2]. The datasets include out-of-equilibrium and near-transition-state structures, respectively, and represent varied energy and force distributions. The ANI-1x dataset consists of Density Functional Theory (DFT) calculations for approximately 5 million diverse molecular conformations with an average of 8 heavy atoms (C, N, O) and an average of 15 total atoms (including H) along with multiple properties including total energy and interatomic forces computed at the ωB97x/6-31G(d) level of theory. The dataset was generated by perturbing equilibrium configurations using an active learning procedure to ensure conformational diversity with the aim of developing an accurate and transferable ML potential. The Transition1x dataset contains DFT calculations of energy and forces, for 9.6 million molecular conformations with up to 7 heavy atoms (C, N, O) and an average of 14 total atoms (including H), likewise computed at the ωB97x/6-31G(d) level of theory. Here, the structures were sampled on and around full reaction pathways, thus including conformations far from equilibrium and near transition states. The dataset was generated by running a nudged elastic band (NEB)[34] algorithm with DFT on a set of approximately 10 thousand organic reactions with up to 6 bond changes while saving intermediate calculations with the aim of improving ML potentials around transition states. Transition1x is more varied in terms of interatomic distances between pairs of heavy atoms than ANI-1x, but less varied in terms of the distribution of forces, since forces are generally minimised along reaction pathways[2,34].

3.2 Model hyperparameters

The same general model configuration was used in all experiments. Each individual graph neural network model was configured with 3 interaction layers, a hidden node state size of 256 and a 2-layer atom-wise readout network with 3 outputs representing E_i, $\sigma^2_{E_i}$ and $\sigma^2_{F_i}$. The input molecular graphs were generated with an edge cutoff radius of 5.0 Å. Models were trained using the Adam optimizer with an initial learning rate of 10^{-4}, an exponential decay learning rate scheduler, a batch size of 64 molecular graphs, force loss weight $\lambda_F = 0.5$ and an early stopping criterion on the validation loss to prevent overfitting. In each experiment, models were trained individually with the same hyperparameters on the same training data, but with different random parameter initialisation and random shuffling of the training data to induce model diversity in the ensembles. For each ensemble model, after the training was completed, a recalibration function was fitted using predictions on the respective validation dataset following the procedure described in Section 2.6 and applied to the predictions on the test data. For ensemble models trained with MSE loss, only the epistemic uncertainty is considered.

Table 1 Test results after recalibration of ensemble models ($M = 5$) trained on the ANI-1x (A1x) and Transition1x (T1x) datasets with different combinations of mean squared error (MSE) and negative log likelihood (NLL) loss functions on the energies and forces. Energy errors are averaged over molecules, while force errors are computed component-wise and averaged over the spatial dimensions and atoms.

Data	Loss		Energy (eV)							Forces (eV/Å)						
	$\mathscr{L}_E$	$\mathscr{L}_F$	MAE↓	RMSE↓	NLL↓	RZV	ENCE↓	CV	RMV	MAE↓	RMSE↓	NLL↓	RZV	ENCE↓	CV	RMV
A1x	MSE	MSE	0.0123	0.0278	-2.81	0.97	0.0773	1.27	0.0283	0.0180	0.0362	-2.56	0.98	0.0243	1.24	0.0386
	NLL	MSE	0.0118	0.0256	-3.04	1.00	0.0411	1.00	0.0209	0.0179	0.0364	-2.56	0.98	0.0229	1.22	0.0381
	MSE	NLL	0.0117	0.0305	-2.91	0.97	0.0928	1.53	0.0305	0.0175	0.0399	-2.77	1.00	0.0099	1.51	0.0410
	NLL	NLL	0.0105	0.0296	**-3.26**	1.02	0.0600	1.51	0.0237	0.0171	0.0402	**-2.79**	1.00	0.0093	1.56	0.0409
T1x	MSE	MSE	0.0344	0.0612	-1.79	0.89	0.1144	0.82	0.0682	0.0370	0.0743	-1.94	0.99	0.0292	1.21	0.0804
	NLL	MSE	0.0318	0.0578	-1.94	0.92	0.1383	0.86	0.0655	0.0366	0.0744	-1.97	0.98	0.0293	1.27	0.0831
	MSE	NLL	0.0332	0.0600	-1.84	0.95	0.1108	0.83	0.0628	0.0369	0.0745	-1.99	0.94	0.0615	1.18	0.0817
	NLL	NLL	0.0303	0.0574	**-2.09**	0.98	0.0906	0.96	0.0562	0.0359	0.0751	**-2.05**	0.94	0.0645	1.15	0.0773

3.3 ANI-1x results

Ensembles of graph neural network models were trained on ANI-1x using the data splits from Schreiner et al. [2], where the validation and test datasets consist of approximately 5% of the data each and the training set consists of the remaining 90% of the data. The splits are stratified by chemical formula to ensure different splits do not contain configurations made up of exactly the same atoms and selected such that all splits include all species of heavy atoms. Individual models were trained for up to 10 million gradient steps (approximately 144 epochs) with an initial warmup period of 2 million steps where the model was trained only with MSE loss followed by an interpolation period of 1 million steps, where the loss was interpolated linearly to NLL loss (only NLL models). The ensemble predictions were then recalibrated post hoc using a recalibration function fitted using the validation dataset. The trade-off between validation performance and ensemble size M using models trained with NLL loss on both energy and forces is illustrated in Figure 1. Using a larger ensemble size results in lower error, as expected, but comes at the cost of additional computations. We observe that a reasonably low error is obtained at $M = 5$ and only small improvements are gained beyond that, which is similar to what we found in previous work [27].
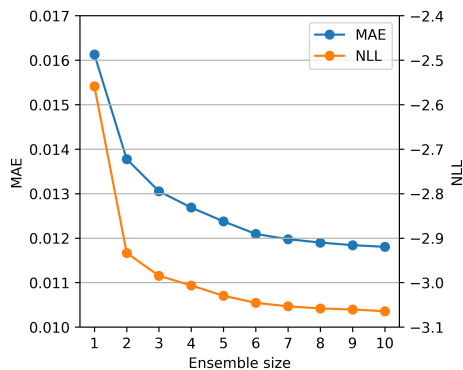

Fig. 1 Trade-off between performance and ensemble size on the ANI-1x validation dataset using ensembles of models trained with NLL loss on both energy and forces.

Test results for $M = 5$ ensembles trained with different combinations of MSE and NLL loss on energy and forces are presented in the first four rows of Table 1. The ensemble trained with MSE loss on both energy and forces is a standard ensemble model with post hoc recalibration. The other three ensembles show the effect of training with NLL loss on either energy or forces or both. All four ensembles achieved a low error on energy and forces in terms of MAE and RMSE compared to the results reported by Schreiner et al. [2] using a PaiNN model similar to the base model of our ensembles (MAE=0.023 on energy and MAE=0.039 on forces). Importantly, training with NLL loss on either energy or forces or both did not result in worse performance in terms of prediction error.

All four ensemble models achieved good average calibration on ANI-1x in terms of NLL and RZV after recalibration, but ensembles trained with NLL loss performed slightly better which is observed both on energy and forces and the ensemble trained with NLL on both energy and forces performs best overall. Additionally, all ensembles scored a high CV indicating uncertainty estimates are heteroscedastic and thus informative. Calibration plots for the ensemble trained on ANI-1x with NLL loss on energy and forces are presented in Figure 2. The uncertainty vs. error plots show that in general large errors are associated with large uncertainties and most errors are within 2-3 standard deviations of uncertainty as desired. For the energy, the model appears to be biased for some examples with large errors, but these are relatively few and are correctly identified as problematic by high uncertainty. For the forces, the distribution of errors looks more symmetrical around zero. This is also clearly shown by the local z-score analysis plots where for the energy, the variance of the z-scores is slightly off for very low and very high uncertainties, although still centered around 1, whereas for the forces the variance of the z-scores is close to 1 for all uncertainties which indicates high consistency. Finally, the reliability diagrams show the relation between predicted uncertainty and expected error for the energy and forces, respectively. Both plots show a clear correlation between the uncertainty and the expected error as the curves lie close to the identity line. Again, the model very slightly underestimates the expected error of the energy at low and high uncertainties, and the curve for the forces is near perfect. The reliability diagrams are summarised by ENCE scores in Table 1. Additional calibration results are included in the ESI.
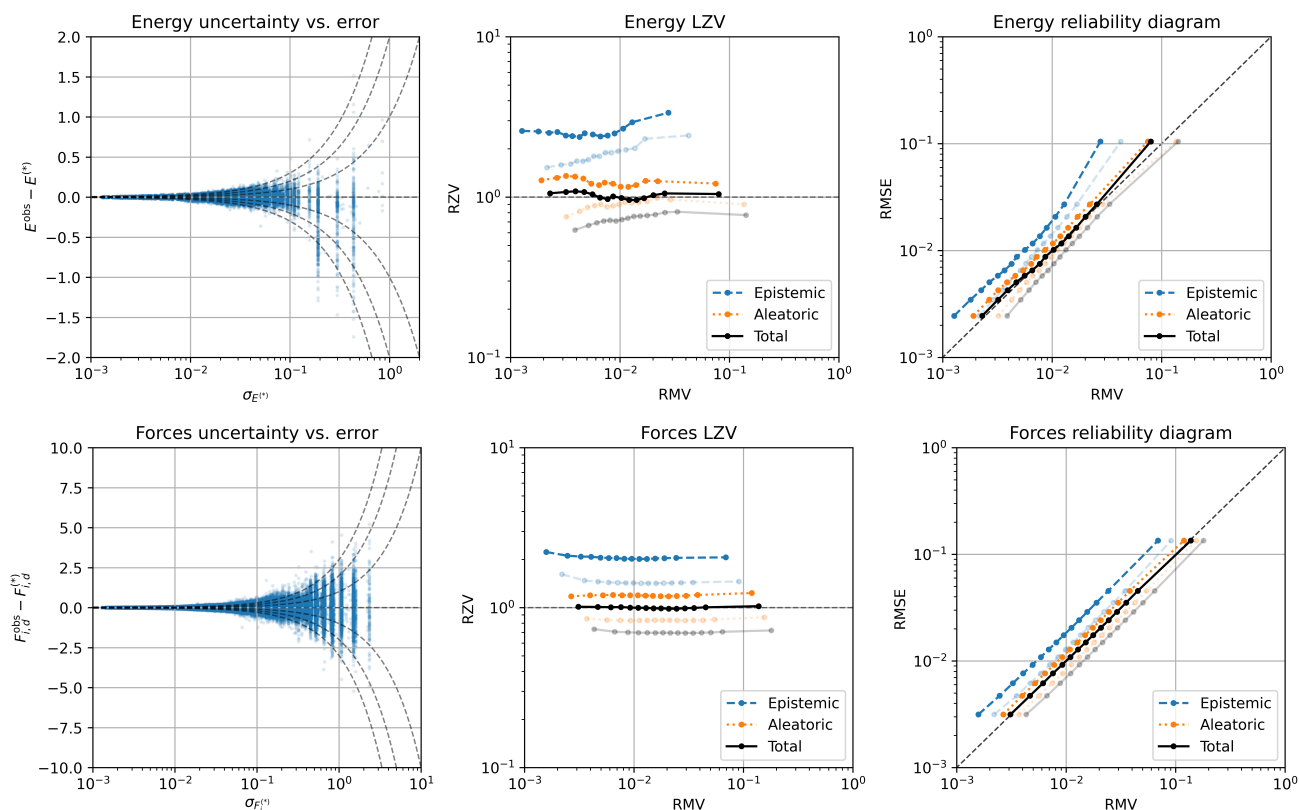
Fig. 2 Calibration results on the ANI-1x dataset of energy (top row) and forces (bottom row) for an ensemble of $M = 5$ models trained with NLL loss on both energy and forces. To illustrate the effect of recalibration, the transparent curves show results before applying recalibration (energy ENCE=0.2650, forces ENCE=0.2964) whereas the solid curves show results after recalibration (energy ENCE=0.0600, forces ENCE=0.0093). The LZV analyses and reliability diagrams are generated using 15 equal sized bins. All curves in each plot use the same bins based on sorting by total uncertainty.

3.4 Transition1x results

Analogous to the first experiment, ensembles of graph neural network models were trained on Transition1x using data splits from Schreiner *et al.* [2] based on the same splitting criteria as ANI-1x described above. Models were trained for up to 3 million gradient steps (approximately 21 epochs) with an initial warmup period of 2 million steps followed by an interpolation period of 2 million steps (training was stopped before finishing the full interpolation period). When training for longer on this dataset, we observed severe overfitting. We believe this is because the data was generated from a relatively small set of chemical reactions making the models prone to overfit the many similar configurations associated with the same reactions in the training data. The ensemble predictions were recalibrated post hoc using a recalibration function fitted using the validation dataset. Varying the ensemble size yielded similar results to the first experiment (Figure 1) and $M = 5$ was selected as a good compromise between performance and computational cost.

Test results for $M = 5$ ensembles are presented in the last four rows of Table 1. As in the first experiment, all ensembles achieved a low error on energy and forces in terms of MAE and RMSE compared to the results reported in Schreiner *et al.* [2] (MAE=0.048 on energy and MAE=0.058 on forces) and training with NLL loss did not decrease performance in terms of prediction error in any

case. All four ensembles achieved acceptable average calibration in terms of NLL and RZV on the Transition1x test data. Surprisingly, ensembles trained with MSE loss were as well or better calibrated than ensembles trained with NLL loss on this dataset. All ensembles score similar high CV indicating uncertainty estimates are heteroscedastic and thus informative. Calibration plots for an ensemble trained on Transition1x with NLL loss on energy and forces are presented in Figure 3. The uncertainty vs. error plots show that in general large errors are associated with large uncertainties as desired. For both energy and forces some errors extend beyond 2-3 standard deviations of uncertainty indicating the error distributions have wider tails and may not be Gaussian in this case. Similar to the ANI-1x experiment, it looks like the model is biased for some instances with large energy errors, but these cases are correctly identified as problematic by high uncertainty. For the forces, the error distribution appears more symmetrical around zero but with wide tails. The local z-score analysis plot for the energy indicate some inconsistencies in the energy uncertainties. Plotting the root variance of the z-scores as a function of the observed molecular energies (Figure 4) shows a tendency of the model to underestimate the uncertainty for low energies and overestimate the uncertainty for high energies on average. This is a problem with the model that can not be corrected by scaling the uncertainties in the recalibration step. Taking a closer look

at the energy distribution reveals significant differences between the training, validation and test sets that is likely a consequence of splitting the data on chemical formula which could be the reason for this problem. The variance of the local z-scores for the forces are more consistent, but values below one indicate that the model generally overestimates the uncertainty on the forces. The reliability diagram for the energy also shows signs of some inconsistencies, as the curve does not form a straight line along the diagonal, but overall the uncertainties are correlated with the expected error. The corresponding reliability diagram for the forces shows a more consistent result, only with a tiny overestimation of the force uncertainty. The reliability diagrams are summarised by ENCE scores in Table 1. Additional calibration results are included in the ESI.

4   Discussion

The proposed method achieved good predictive performance as well as calibrated and consistent uncertainty estimates in experiments on two challenging, publicly available molecular datasets. A major advantage of the approach is that it considers both epistemic and aleatoric uncertainty through an ensemble approximation of mean-variance models. We believe that considering both aleatoric and epistemic uncertainty is critical to ensure good calibration in and out of the training data distribution. Often the training procedures of uncertainty aware models do not inherently ensure good calibration on unseen data. For example, ensemble members trained on the same data will often fit the same mean prediction without accounting for errors caused by random noise or inconsistency in the data or model inadequacy and mean-variance methods will estimate the expected error on the training data but do not guarantee good extrapolation of the uncertainty estimates to unseen data. Therefore, the post hoc recalibration procedure is key to achieve good calibration on unseen data in our experiments, but is not commonly applied by other UQ methods in the literature.

The computational overhead of training and evaluating ensemble models is sometimes pointed out as a major disadvantage of using ensembles. However, it is important to note that most of this computation can be performed in parallel and thus only leads to a small overhead of computing the ensemble approximation and recalibration in real time. Some works have proposed methods for speeding up the training of ensembles, such as snapshot ensembles[35,36], which could also be applied in this case. Another widely accepted advantage of ensembles is that they often improve prediction accuracy (see Figure 1 as an example), which can be considered a positive side effect of the proposed method. Here, we have used ensembles of size 5, but larger ensembles can be expected to further improve performance (up to a limit) at the cost of more computation. The approach could also potentially benefit from other recent extensions to ensembles such as using randomized priors[37] to improve the quality of especially the epistemic uncertainty estimates.

Evaluation of uncertainty calibration for regression models is an active area of research[20–23]. Standard procedures for assessing the quality of uncertainty estimates are necessary within the field to establish confidence in individual UQ methods and ensure fair comparison. We recommend recent work by Pernot[38] which provides a good overview of calibration assessment methods and a detailed approach for evaluating uncertainty. Our experiments show that the uncertainty estimates obtained with the proposed method are largely consistent with the expected error for varying size of the uncertainty (Figures 2 and 3). However, we observed indications that uncertainties are not equally well calibrated along different molecular energies (Figure 4). The current recalibration method only considers the magnitude of the predicted uncertainty. It would be an interesting direction for future work to design a recalibration function that can account for additional input features such as the (predicted) energy, while remaining a monotonic increasing scaling function, with the aim of achieving equally good calibration throughout the input space. Applying the calibration evaluation framework proposed by Pernot[38] could help provide additional insights into the consistency and adaptivity of predictive uncertainty.

5   Conclusion

In this work, we have presented a complete framework for training neural network potentials with calibrated uncertainty estimates on both energy and forces. The proposed method was demonstrated and evaluated on two challenging, publicly available molecular datasets containing diverse conformations far from equilibrium. In all cases, the proposed method achieved low prediction error and good uncertainty calibration. On the ANI-1x dataset training with NLL loss improved the calibration over training with standard MSE loss. On the Transition1x dataset, the same improvement was not observed and good calibration was achieved by training with standard MSE loss and applying post hoc nonlinear recalibration. This could be because the validation and test data are more out of distribution in this case. The proposed method does not depend on the particular architecture of the neural network model, and can thus easily be adapted to new models in the future. We hope that this work will contribute to better calibrated ML potentials and enable more robust and reliable applications.

Author Contributions

JB and PBJ: Conceptualization, Methodology, Software, Validation. **JB:** Writing - Original Draft, Visualization. **MNS, OW, TV and PBJ:** Writing - Review & Editing.

Conflicts of interest

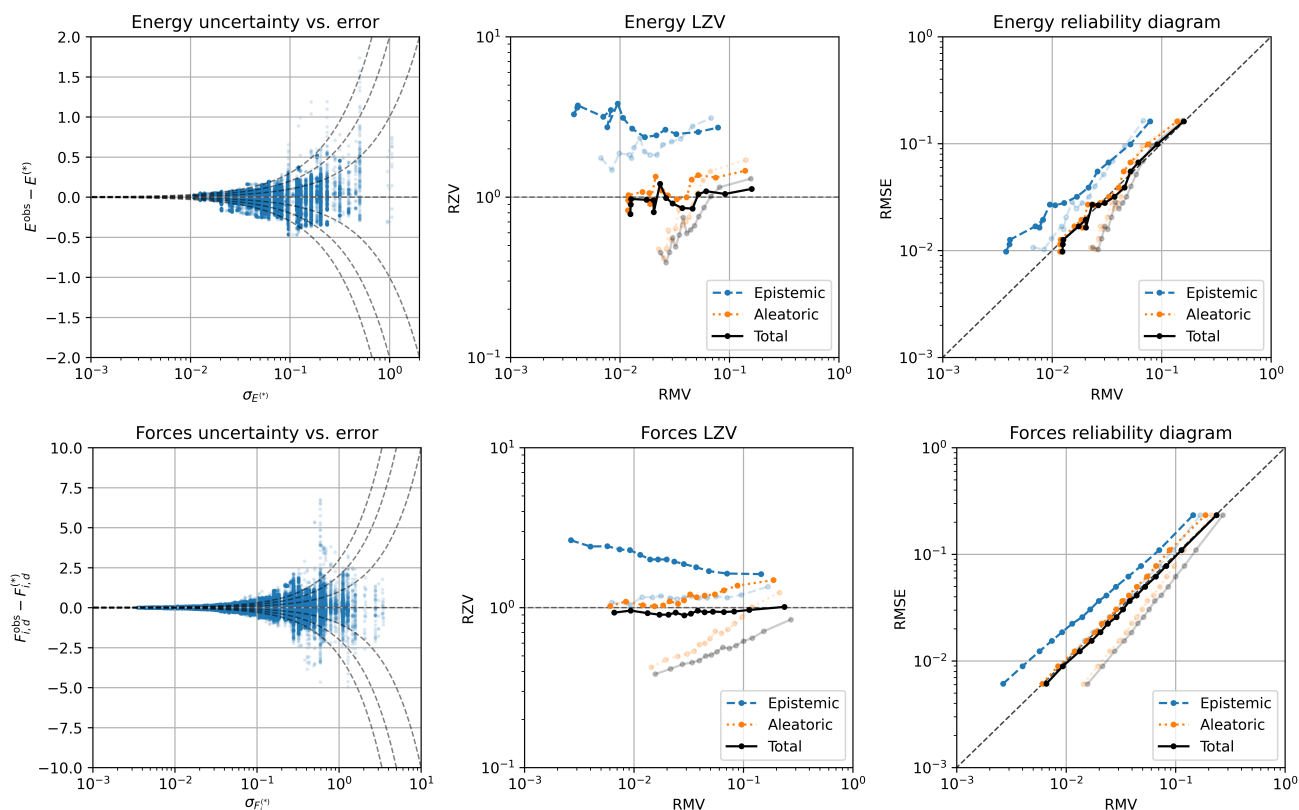There are no conflicts to declare.

Acknowledgements

Fig. 3 Calibration results on the Transition1x dataset of energy (top row) and forces (bottom row) for an ensemble of $M = 5$ models trained with NLL loss on both energy and forces. To illustrate the effect of recalibration, the transparent curves show results before applying recalibration (energy ENCE=0.3339, forces ENCE=0.4502) whereas the solid curves show results after recalibration (energy ENCE=0.0906, forces ENCE=0.0645). The LZV analyses and reliability diagrams are generated using 15 equal sized bins. All curves in each plot use the same bins based on sorting by total uncertainty.
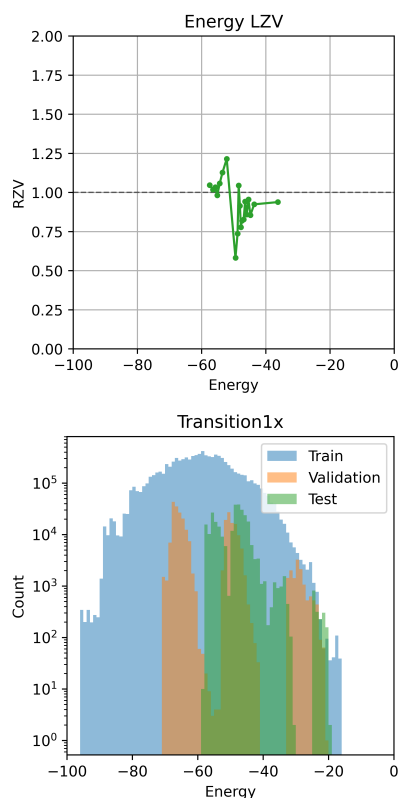

Fig. 4 Local root z-score variance (RZV) conditioned on observed energy on the Transition1x test dataset (top). Energy distribution in the Transition1x data split (bottom).

References

1 J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev and A. E. Roitberg, *The Journal of Chemical Physics*, 2018, **148**, 241733.

2 M. Schreiner, A. Bhowmik, T. Vegge, J. Busk and O. Winther, *Scientific Data*, 2022, **9**, 779.

3 P. O. Dral, *J. Phys. Chem. Lett.*, 2020, **11**, 2336–2347.

4 O. A. von Lilienfeld and K. Burke, *Nature Communications*, 2020, **11**, 4895.

5 A. A. Peterson, R. Christensen and A. Khorshidi, *Phys. Chem. Chem. Phys.*, 2017, **19**, 10978–10985.

6 E. Hüllermeier and W. Waegeman, *Machine Learning*, 2021, **110**, 457–506.

7 A. Kendall and Y. Gal, Proceedings of the 31st International Conference on Neural Information Processing Systems, Red Hook, NY, USA, 2017, p. 5580–5590.

8 C. Schran, K. Brezina and O. Marsalek, *The Journal of Chemical Physics*, 2020, **153**, 104105.

9 A. G. Wilson and P. Izmailov, Proceedings of the 34th Interna-

tional Conference on Neural Information Processing Systems, Red Hook, NY, USA, 2020, pp. 4697–4708.

10 L. Hoffmann and C. Elster, *Deep Ensembles from a Bayesian Perspective*, 2021.

11 F. K. Gustafsson, M. Danelljan and T. B. Schon, 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Los Alamitos, CA, USA, 2020, pp. 1289–1298.

12 R. Neal, Advances in Neural Information Processing Systems, 1992.

13 C. Blundell, J. Cornebise, K. Kavukcuoglu and D. Wierstra, Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37, 2015, p. 1613–1622.

14 Y. Gal and Z. Ghahramani, Proceedings of The 33rd International Conference on Machine Learning, New York, New York, USA, 2016, pp. 1050–1059.

15 D. A. Nix and A. S. Weigend, Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94), 1994, pp. 55–60 vol.1.

16 A. Amini, W. Schwarting, A. Soleimany and D. Rus, Advances in Neural Information Processing Systems, 2020, pp. 14927–14937.

17 A. P. Soleimany, A. Amini, S. Goldman, D. Rus, S. N. Bhatia and C. W. Coley, *ACS Central Science*, 2021, **7**, 1356–1367.

18 Y. Hu, J. Musielewicz, Z. W. Ulissi and A. J. Medford, *Machine Learning: Science and Technology*, 2022, **3**, 045028.

19 B. Lakshminarayanan, A. Pritzel and C. Blundell, *Advances in neural information processing systems*, 2017, **30**, 6402–6413.

20 V. Kuleshov, N. Fenner and S. Ermon, Proceedings of the 35th International Conference on Machine Learning, 2018, pp. 2796–2804.

21 D. Levi, L. Gispan, N. Giladi and E. Fetaya, *Sensors*, 2022, **22**,.

22 P. Pernot, *The Journal of Chemical Physics*, 2022, **157**, 144103.

23 K. Tran, W. Neiswanger, J. Yoon, Q. Zhang, E. Xing and Z. W. Ulissi, *Machine Learning: Science and Technology*, 2020, **1**, 025006.

24 L. Hirschfeld, K. Swanson, K. Yang, R. Barzilay and C. W. Coley, *Journal of Chemical Information and Modeling*, 2020, **60**, 3770–3780.

25 G. Scalia, C. A. Grambow, B. Pernici, Y.-P. Li and W. H. Green, *Journal of Chemical Information and Modeling*, 2020, **60**, 2697–2717.

26 A. Nigam, R. Pollice, M. F. D. Hurley, R. J. Hickman, M. Aldeghi, N. Yoshikawa, S. Chithrananda, V. A. Voelz and A. Aspuru-Guzik, *CoRR*, 2021, **abs/2102.11439**,.

27 J. Busk, P. B. Jørgensen, A. Bhowmik, M. N. Schmidt, O. Winther and T. Vegge, *Machine Learning: Science and Technology*, 2021, **3**, 015012.

28 K. Schütt, O. Unke and M. Gastegger, International Conference on Machine Learning, 2021, pp. 9377–9388.

29 N. Skafte, M. Jørgensen and S. Hauberg, Advances in Neural Information Processing Systems, 2019.

30 M. Seitzer, A. Tavakoli, D. Antic and G. Martius, International Conference on Learning Representations, 2022.

31 E. T. Jaynes, *Phys. Rev.*, 1957, **106**, 620–630.

32 D. D. J. Blower, *Information Processing: The Maximum Entropy Principle*, CreateSpace Independent Publishing Platform, 2013.

33 F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot and E. Duchesnay, *Journal of Machine Learning Research*, 2011, **12**, 2825–2830.

34 D. Sheppard, R. Terrell and G. Henkelman, *The Journal of Chemical Physics*, 2008, **128**,.

35 G. Huang, Y. Li, G. Pleiss, Z. Liu, J. E. Hopcroft and K. Q. Weinberger, *Snapshot Ensembles: Train 1, get M for free*, 2017.

36 F. Wang, G. Wei, Q. Liu, J. Ou, H. Lv *et al.*, *Advances in Neural Information Processing Systems*, 2021, **34**, 19719–19729.

37 I. Osband, J. Aslanides and A. Cassirer, Advances in Neural Information Processing Systems, 2018, pp. 8617–8629.

38 P. Pernot, *Validation of uncertainty quantification metrics: a primer based on the consistency and adaptivity concepts*, 2023.

Electronic Supplementary Information:

Graph Neural Network Interatomic Potential Ensembles with Calibrated Aleatoric and Epistemic Uncertainty on Energy and Forces

Jonas Busk (jbusk@dtu.dk), Mikkel N. Schmidt, Ole Winther, Tejs Vegge and Peter Bjørn Jørgensen

A   Additional results

Here we include additional calibration results from the experiments on the ANI-1x and Transition1x datasets presented in Section 3 of the main paper.

Confidence curves can be used to evaluate the ranking ability of the model and to estimate the drop in error as a percentage of the high uncertainty instances are removed [17,25], which is especially important in applications such as active learning where high uncertainty instances are iteratively added to the training set to improve the model. A confidence curve is generated by sorting predictions by uncertainty in decreasing order and computing the error as a function of removing a percentage of the most uncertain predictions. For a well calibrated model the confidence curve is expected to decrease monotonically. An oracle curve representing perfect ranking can be generated by sorting the prediction by error instead and the confidence curve can be summarised by computing the area between the confidence and oracle curves (AUCO). However, we do not expect the uncertainty predictions to produce a perfect ranking with respect to the errors since instances with high predicted uncertainty can still have small empirical errors.

Quantile-calibration plots compare the quantiles of the predictive distribution with the quantiles of the empirical distribution and is a way to evaluate distribution calibration averaged over the data [20]. If the predictive distribution matches the empirical distribution, the quantile-calibration curve corresponds to the identity function and forms a line along the diagonal of the plot. Assuming a symmetric distribution, the confidence interval can be evaluated instead of the quantile. The quantile-calibration curve can be summarised by the sum of squared errors (SSE) between the predicted and empirical quantiles.

To further check the distribution assumptions averaged over the data, a histogram the errors normalised by the predicted uncertainties (z-scores) along with the assumed standard distribution can also be plotted.

A.1   Additional ANI-1x results

Additional calibration plots for the ensemble model trained on ANI-1x with NLL loss on energy and forces are presented in Figure A.1. The confidence curves for energy and forces are both monotonically decreasing indicating good ranking ability. The confidence curves also show a significant drop in error on both energy and forces when removing the top $\sim 5\%$ highest uncertainty predictions. This confirms the observation from the reliability diagrams in Figure 2, that there are a few instances with very high error but they are correctly identified and assigned high uncertainty by the model.

The energy quantile-calibration plot shows that assuming a normal distribution percentiles of the predicted distributions corresponds well to the empirical distribution and the symmetry at the 0.5 percentile indicates that the model is unbiased overall. In the case of the forces, we assume the distribution of the component-wise errors is normal and unbiased. Furthermore, if the component-wise force errors are normally distributed, the squared L2 norm of the 3-dimensional force errors should follow a chi-square distribution with 3 degrees of freedom. Consequently, we plot the symmetric version of the quantile-calibration plot for the component-wise force errors using a normal distribution and a regular quantile plot for the squared L2 norm of the force errors using a chi-square distribution as they allow for easier comparison. In both cases, the uncertainty estimates look fairly well calibrated with regards to the assumed distributions. This is also apparent from the corresponding histograms of normalised errors plotted along with the reference distributions.

A.2   Additional Transition1x results

Additional calibration plots for the ensemble model trained on Transition1x with NLL loss on energy and forces are presented in Figure A.2. The energy confidence curve is generally decreasing, but like the corresponding reliability diagram shown in Figure 3 it is not perfectly consistent whereas the forces confidence curve look more consistent and monotonically decreasing. In both cases, removing the highest uncertainty instances results in a large drop en error.

The energy quantile-calibration plot shows that the error is fairly well distribution calibrated with a slight overestimation of the uncertainty on average. The same applies to the forces where the error is also fairly well distribution calibrated but with some overestimation of the uncertainty. This is consistent with the results of the LZV analysis described in Section 3.4.
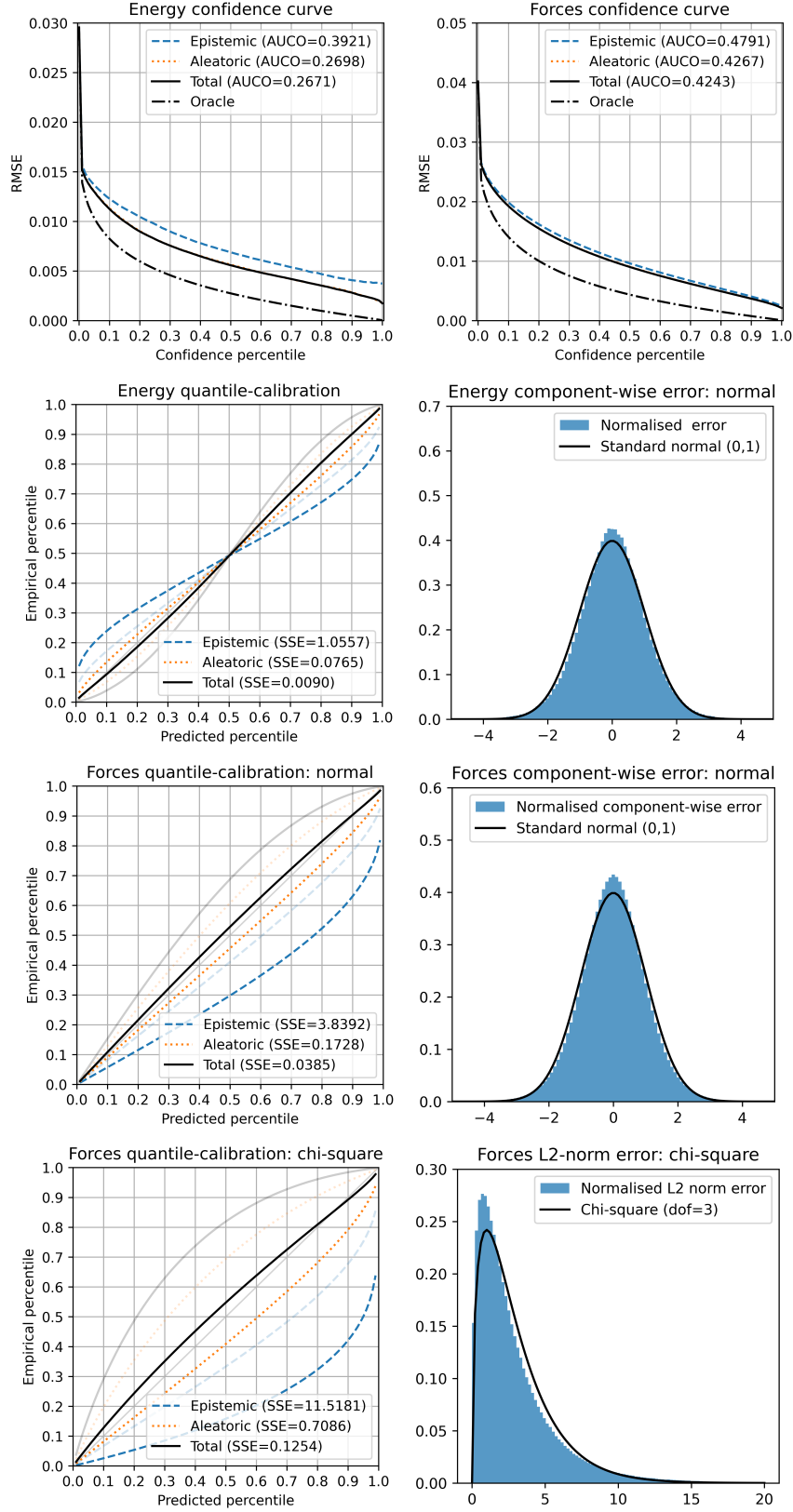
Fig. A.1 Additional calibration results on the ANI-1x dataset of energy and forces for an ensemble of $M = 5$ models trained with NLL loss on both energy and forces. To illustrate the effect of recalibration, the transparent curves show results before applying recalibration whereas the solid curves show results after recalibration.
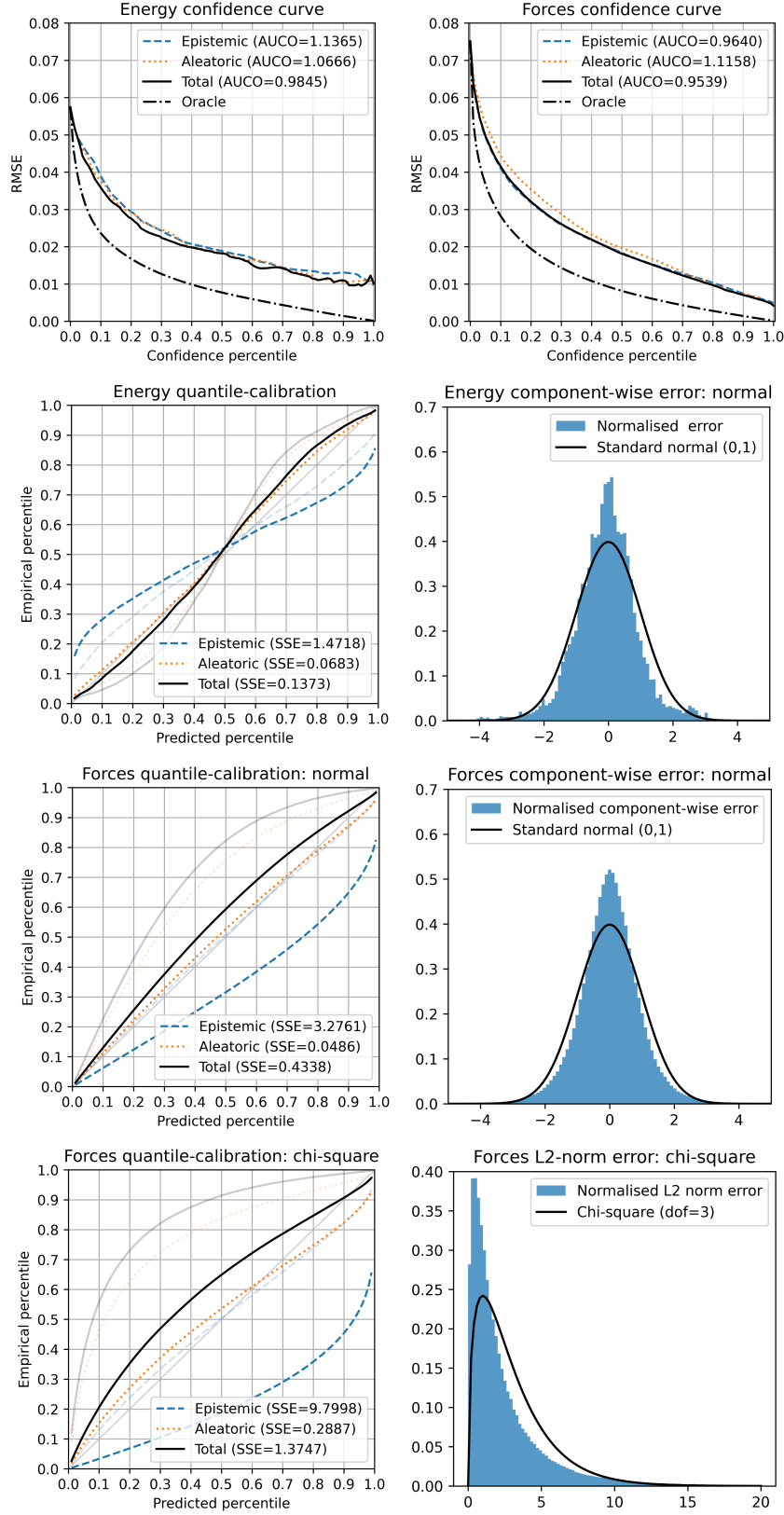
Fig. A.2 Additional calibration results on the Transition1x dataset of energy and forces for an ensemble of $M = 5$ models trained with NLL loss on both energy and forces. To illustrate the effect of recalibration, the transparent curves show results before applying recalibration whereas the solid curves show results after recalibration.