

## SchNet – A deep learning architecture for molecules and materials

K. T. Schütt, H. E. Sauceda, P.-J. Kindermans, A. Tkatchenko, and K.-R. Müller

Citation: *The Journal of Chemical Physics* **148**, 241722 (2018); doi: 10.1063/1.5019779

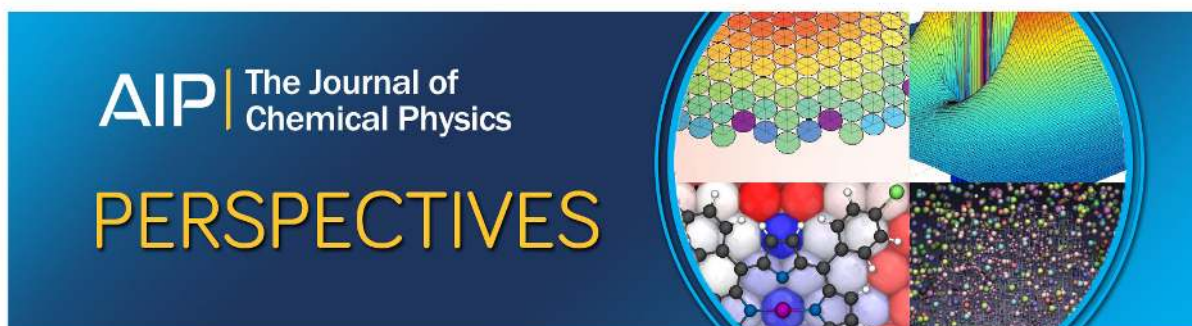
View online: <https://doi.org/10.1063/1.5019779>

View Table of Contents: <http://aip.scitation.org/toc/jcp/148/24>

Published by the [American Institute of Physics](#)

---

---



# SchNet – A deep learning architecture for molecules and materials

K. T. Schütt,<sup>1,a)</sup> H. E. Sauceda,<sup>2</sup> P.-J. Kindermans,<sup>1</sup> A. Tkatchenko,<sup>3,b)</sup> and K.-R. Müller<sup>1,4,5,c)</sup>

<sup>1</sup>Machine Learning Group, Technische Universität Berlin, 10587 Berlin, Germany

<sup>2</sup>Fritz-Haber-Institut der Max-Planck-Gesellschaft, 14195 Berlin, Germany

<sup>3</sup>Physics and Materials Science Research Unit, University of Luxembourg, L-1511 Luxembourg, Luxembourg

<sup>4</sup>Max-Planck-Institut für Informatik, Saarbrücken, Germany

<sup>5</sup>Department of Brain and Cognitive Engineering, Korea University, Anam-dong, Seongbuk-gu, Seoul 136-713, South Korea

(Received 16 December 2017; accepted 8 March 2018; published online 29 March 2018)

Deep learning has led to a paradigm shift in artificial intelligence, including web, text, and image search, speech recognition, as well as bioinformatics, with growing impact in chemical physics. Machine learning, in general, and deep learning, in particular, are ideally suitable for representing quantum-mechanical interactions, enabling us to model nonlinear potential-energy surfaces or enhancing the exploration of chemical compound space. Here we present the deep learning architecture SchNet that is specifically designed to model atomistic systems by making use of continuous-filter convolutional layers. We demonstrate the capabilities of SchNet by accurately predicting a range of properties across chemical space for *molecules and materials*, where our model learns chemically plausible embeddings of atom types across the periodic table. Finally, we employ SchNet to predict potential-energy surfaces and energy-conserving force fields for molecular dynamics simulations of small molecules and perform an exemplary study on the quantum-mechanical properties of C<sub>20</sub>-fullerene that would have been infeasible with regular *ab initio* molecular dynamics. *Published by AIP Publishing.* <https://doi.org/10.1063/1.5019779>

## I. INTRODUCTION

Accelerating the discovery of molecules and materials with desired properties is a long-standing challenge in computational chemistry and the materials sciences. However, the computational cost of accurate quantum-chemical calculations proves prohibitive in the exploration of the vast chemical space. In recent years, there have been increased efforts to overcome this bottleneck using machine learning, where only a reduced set of reference calculations is required to accurately predict chemical properties<sup>1–15</sup> or potential-energy surfaces.<sup>16–25</sup> While these approaches make use of painstakingly handcrafted descriptors, deep learning has been applied to predict properties from molecular structures using graph neural networks.<sup>26,27</sup> However, these are restricted to predictions for equilibrium structures due to the lack of atomic positions in the input. Only recently, approaches that learn a representation directly from atom types and positions have been developed.<sup>28–30</sup> While neural networks are often considered as a “black box,” there has recently been an increased effort to explain their predictions in order to understand how they operate or even extract scientific insight. This can be done either by analyzing a trained model<sup>31–37</sup> or by directly designing interpretable models.<sup>38</sup> For quantum chemistry, some of us have proposed such an interpretable architecture with Deep Tensor

Neural Networks (DTNNs) that not only learns the representation of atomic environments but also allows for spatially and chemically resolved insights into quantum-mechanical observables.<sup>28</sup>

Here we build upon this work and present the deep learning architecture SchNet that allows us to model complex atomic interactions in order to predict potential-energy surfaces or speeding up the exploration of chemical space. SchNet, being a variant of DTNNs, is able to learn representations for molecules and materials that follow fundamental symmetries of atomistic systems by construction, e.g., rotational and translational invariance as well as invariance to atom indexing. This enables accurate predictions throughout compositional and configurational chemical space, where symmetries of the potential energy surface are captured by design. Interactions between atoms are modeled using continuous-filter convolutional (cfconv) layers<sup>30</sup> being able to incorporate further chemical knowledge and constraints using specifically designed filter-generating neural networks. We demonstrate that these allow us to efficiently incorporate periodic boundary conditions (PBCs) enabling accurate predictions of formation energies for a diverse set of bulk crystals. Beyond that, both SchNet and DTNNs provide local chemical potentials to analyze the obtained representation and allow for chemical insights.<sup>28</sup> An analysis of the obtained representation shows that SchNet learns chemically plausible embeddings of atom types that capture the structure of the periodic table. Finally, we present a path-integral molecular dynamics (PIMD) simulation using an energy-conserving force field learned by SchNet trained on reference data from a classical MD at

<sup>a)</sup>kristof.schuett@tu-berlin.de

<sup>b)</sup>alexandre.tkatchenko@uni.lu

<sup>c)</sup>klaus-robert.mueller@tu-berlin.de

the PBE+vdW<sup>TS39,40</sup> level of theory effectively accelerating the simulation by three orders of magnitude. Specifically, we employ the recently developed perturbed path-integral approach<sup>41</sup> for carrying out imaginary time PIMD, which allows quick convergence of quantum-mechanical properties with respect to the number of classical replicas (beads). This exemplary study shows the advantages of developing computationally efficient force fields with *ab initio* accuracy, allowing nanoseconds of PIMD simulations at low temperatures—an inconceivable task for regular *ab initio* molecular dynamics (AIMD) that could be completed with SchNet within hours instead of years.

## II. METHOD

SchNet is a variant of the earlier proposed Deep Tensor Neural Networks (DTNNs)<sup>28</sup> and therefore shares a number of their essential building blocks. Among these are atom embeddings, interaction refinements and atom-wise energy contributions. At each layer, the atomistic system is represented atom-wise being refined using pairwise interactions with the surrounding atoms. In the DTNN framework, interactions are modeled by tensor layers, i.e., atom representations and interatomic distances are combined using a parameter tensor. This can be approximated using a low-rank factorization for computational efficiency.<sup>42–44</sup> SchNet instead makes use of continuous-filter convolutions with filter-generating networks<sup>30,45</sup> to model the interaction term. These can be interpreted as a special case of such factorized tensor layers. In the following, we introduce these components and describe how they are assembled to form the SchNet architecture. For an overview of the SchNet architecture, see Fig. 1.

### A. Atom embeddings

An atomistic system can be described uniquely by a set of  $n$  atom sites with nuclear charges  $Z = (Z_1, \dots, Z_n)$  and positions  $R = (\mathbf{r}_1, \dots, \mathbf{r}_n)$ . Through the layers of SchNet, the atoms are described by a tuple of features  $X^l = (\mathbf{x}_1^l, \dots, \mathbf{x}_n^l)$ , with  $\mathbf{x}_i^l \in \mathbb{R}^F$  with the number of feature maps  $F$ , the number

of atoms  $n$ , and the current layer  $l$ . The representation of site  $i$  is initialized using an embedding dependent on the atom type  $Z_i$

$$\mathbf{x}_i^0 = \mathbf{a}_{Z_i}. \quad (1)$$

These embeddings  $\mathbf{a}_Z$  are initialized randomly and optimized during training. They represent atoms of a system disregarding any information about their environment for now.

### B. Atom-wise layers

Atom-wise layers are dense layers that are applied separately to the representations  $\mathbf{x}_i^l$  of each atom  $i$

$$\mathbf{x}_i^{l+1} = W^l \mathbf{x}_i^l + \mathbf{b}^l. \quad (2)$$

Since weights  $W^l$  and biases  $\mathbf{b}^l$  are shared across atoms, our architecture remains scalable with respect to the number of atoms. While the atom representations are passed through the network, these layers transform them and process information about the atomic environments incorporated through interaction layers.

### C. Interaction blocks

The interaction blocks of SchNet add refinements to the atom representation based on pairwise interactions with the surrounding atoms. In contrast to DTNNs, here we model these with continuous-filter convolutional (cfconv) layers that are a generalization of the discrete convolutional layers commonly used, e.g., for images<sup>46,47</sup> or audio data.<sup>48</sup> This generalization is necessary since atoms are not located on regular grid-like image pixels, but can be located at arbitrary positions. Therefore, a filter-tensor, as used in conventional convolutional layers, is not applicable. Instead we need to model the filters continuously with a filter-generating neural network. Given atom-wise representations  $X^l$  at positions  $R$ , we obtain the interactions of atom  $i$  as the convolution with all surrounding atoms

$$\mathbf{x}_i^{l+1} = (X^l * W^l)_i = \sum_{j=0}^{n_{\text{atoms}}} \mathbf{x}_j^l \circ W^l(\mathbf{r}_j - \mathbf{r}_i), \quad (3)$$

where “ $\circ$ ” represents the element-wise multiplication. Note that we perform feature-wise convolutions for computational efficiency.<sup>49</sup> Cross-feature processing is subsequently performed by atom-wise layers. Instead of a filter tensor, we define a filter-generating network  $W^l: \mathbb{R}^3 \rightarrow \mathbb{R}^F$  that maps the atom positions to the corresponding values of the filter bank (see Sec. II D).

A cfconv layer together with three atom-wise layers constitutes the residual mapping<sup>50</sup> of an interaction block (see Fig. 1, right). We use a shifted softplus  $\text{sps}(x) = \ln(0.5e^x + 0.5)$  as activation functions throughout the network. The shifting ensures that  $\text{sps}(0) = 0$  and improves the convergence of the network while having infinite order of continuity. This allows us to obtain smooth potential energy surfaces, force fields, and second derivatives that are required for training with forces as well as the calculation of vibrational modes.

### D. Filter-generating networks

The filter-generating network determines how interactions between atoms are modeled and can be used to constrain

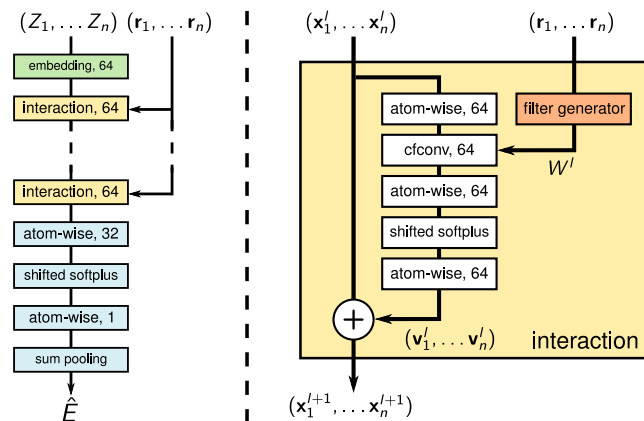


FIG. 1. Illustrations of the SchNet architecture (left) and interaction blocks (right) with atom embedding in green, interaction blocks in yellow, and property prediction network in blue. For each parameterized layer, the number of neurons is given. The filter-generating network (orange) is shown in detail in Fig. 2.

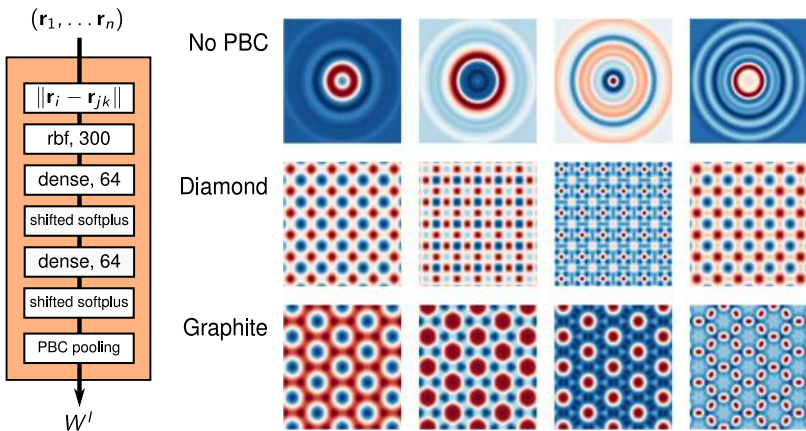


FIG. 2. Architecture of the filter-generating network used in SchNet (left) and  $5 \text{ \AA} \times 5 \text{ \AA}$  cuts through generated filters (right) from the same filter-generating networks (columns) under different periodic bounding conditions (rows). Each filter is learned from data and represents the effect of an interaction on a given feature of an atom representation located in the center of the filter. For each parameterized layer, the number of neurons is given.

the model and include chemical knowledge. We choose a fully connected neural network that takes the vector pointing from atom  $i$  to its neighbor  $j$  as an input to obtain the filter values  $W(\mathbf{r}_j - \mathbf{r}_i)$  (see Fig. 2, left). This allows us to include known invariances of molecules and materials into the model.

### 1. Rotational invariance

It is straightforward to include rotational invariance by computing pairwise distances instead of using relative positions. We further expand the distances in a basis of Gaussians

$$e_k(\mathbf{r}_j - \mathbf{r}_i) = \exp(-\gamma(\|\mathbf{r}_j - \mathbf{r}_i\| - \mu_k)^2),$$

with centers  $\mu_k$  chosen on a uniform grid between zero and the distance cutoff. This has the effect of decorrelating the filter values, which improves the conditioning of the optimization problem. The number of Gaussians and the hyper parameter  $\gamma$  determine the resolution of the filter. We have set the grid spacing and scaling parameter  $\gamma$  to be  $0.1 \text{ 1/\AA}^2$  for all models in this work.

### 2. Periodic boundary conditions

For atomistic systems with periodic boundary conditions (PBCs), each atom-wise feature vector  $\mathbf{x}_i$  has to be equivalent across all periodic repetitions, i.e.,  $\mathbf{x}_i = \mathbf{x}_{ia} = \mathbf{x}_{ib}$  for repeated unit cells  $a$  and  $b$ . Due to the linearity of the convolution, we are, therefore, able to apply the PBCs directly to the filter to accurately describe the atom interactions while keeping invariance to the choice of the unit cell. Given a filter  $\tilde{W}^l(\mathbf{r}_{jb} - \mathbf{r}_{ia})$  over all atoms with  $\|\mathbf{r}_{jb} - \mathbf{r}_{ia}\| < r_{\text{cut}}$ , we obtain the convolution

$$\begin{aligned} \mathbf{x}_i^{l+1} &= \mathbf{x}_{im}^{l+1} = \frac{1}{n_{\text{neighbors}}} \sum_{\substack{j,n \\ \mathbf{r}_{jn}}} \mathbf{x}_{jn}^l \circ \tilde{W}^l(\mathbf{r}_{jn} - \mathbf{r}_{im}) \\ &= \frac{1}{n_{\text{neighbors}}} \sum_j \mathbf{x}_j^l \circ \underbrace{\left( \sum_n \tilde{W}^l(\mathbf{r}_{jn} - \mathbf{r}_{im}) \right)}_W. \end{aligned}$$

This new filter  $W$  now depends on the PBCs of the system as we sum over all periodic images within the given cutoff

$r_{\text{cut}}$ . We find that the training is more stable when normalizing the filter response  $\mathbf{x}_i^{l+1}$  by the number of atoms within the cutoff range. Figure 2 (right) shows a selection of generated filters without PBCs, with a cubic diamond crystal cell and with a hexagonal graphite cell. As the filters for diamond and graphite are superpositions of single-atom filters according to their respective lattice, they reflect the structure of the lattice. Note that while the single-atom filters are circular due to the rotational invariance, the periodic filters become rotationally equivariant with respect to the orientation of the lattice, which still keeps the property prediction rotationally invariant. While we have followed a data-driven approach where we only incorporate basic invariances in the filters, careful design of the filter-generating network provides the possibility to incorporate further chemical knowledge in the network.

### E. Property prediction

Finally, a given property  $P$  of a molecule or material is predicted from the obtained atom-wise representations. We compute atom-wise contributions  $\hat{P}_i$  from the fully connected prediction network (see blue layers in Fig. 1). Depending on whether the property is intensive or extensive, we calculate the final prediction  $\hat{P}$  by summing or averaging over the atomic contributions, respectively.

Since the initial atom embeddings are obviously equivariant to the order of atoms, atom-wise layers are independently applied to each atom and continuous-filter convolutions sum over all neighboring atoms, indexing equivariance is retained in the atom-wise representations. Therefore, the prediction of properties as a sum over atom-wise contributions guarantees indexing invariance.

When predicting atomic forces, we instead differentiate a SchNet predicting the energy with respect to the atomic positions

$$\hat{\mathbf{F}}_i(Z_1, \dots, Z_n, \mathbf{r}_1, \dots, \mathbf{r}_n) = -\frac{\partial \hat{E}}{\partial \mathbf{r}_i}(Z_1, \dots, Z_n, \mathbf{r}_1, \dots, \mathbf{r}_n). \quad (4)$$

When using a rotationally invariant energy model, this ensures rotationally equivariant force predictions and guarantees an energy conserving force field.<sup>21</sup>



## F. Training

We train SchNet for each property target  $P$  by minimizing the squared loss

$$\ell(\hat{P}, P) = \|P - \hat{P}\|^2.$$

For the training of energies and forces of molecular dynamics trajectories, we use a combined loss

$$\ell((\hat{E}, \hat{\mathbf{F}}_1, \dots, \hat{\mathbf{F}}_n), (E, \mathbf{F}_1, \dots, \mathbf{F}_n)) = \rho \|E - \hat{E}\|^2 + \frac{1}{n_{\text{atoms}}} \sum_{i=0}^{n_{\text{atoms}}} \left\| \mathbf{F}_i - \left( -\frac{\partial \hat{E}}{\partial \mathbf{R}_i} \right) \right\|^2, \quad (5)$$

where  $\rho$  is a trade-off between energy and force loss.<sup>51</sup>

All models are trained with mini-batch stochastic gradient descent using the ADAM optimizer<sup>52</sup> with mini-batches of 32 examples. We decay the learning rate exponentially with ratio 0.96 every 100 000 steps. In each experiment, we split the data into a training set of given size  $N$  and use a validation set for early stopping. The remaining data are used for computing the test errors. Since there are a maximum number of atoms being located within a given cutoff, the computational cost of a training step scales linearly with the system size if we precompute the indices of nearby atoms.

## III. RESULTS

### A. Learning molecular properties

We train SchNet models to predict various properties of the QM9 dataset<sup>53–55</sup> of 131k small organic molecules with up to nine heavy atoms from CONF. Following Gilmer *et al.*<sup>29</sup> and Faber *et al.*,<sup>10</sup> we use a validation set of 10 000 molecules. We sum over atomic contribution  $\hat{P}_i$  for all properties but  $\epsilon_{\text{HOMO}}$ ,  $\epsilon_{\text{LUMO}}$ , and the gap  $\Delta\epsilon$ , where we take the average. We use  $T = 6$  interaction blocks and atomic representations with  $F = 64$  feature dimension and perform up to  $10 \times 10^6$  gradient descent parameter updates. Since the molecules of QM9 are quite small, we do not use a distance cutoff. For the Gaussian expansion, we use a range up to 20 Å to cover all interatomic distances occurring in the data. The prediction errors are listed in Table I, where we compare the performance

TABLE I. Mean absolute errors for energy predictions on the QM9 dataset using 110k training examples. For SchNet, we give the average over three repetitions as well as standard errors of the mean of the repetitions. Best models in bold.

Property	Unit	SchNet ( $T = 6$ )	enn-s2s <sup>29</sup>
$\epsilon_{\text{HOMO}}$	eV	<b>0.041 ± 0.001</b>	0.043
$\epsilon_{\text{LUMO}}$	eV	<b>0.034 ± 0.000</b>	0.037
$\Delta\epsilon$	eV	<b>0.063 ± 0.000</b>	0.069
ZPVE	meV	1.7 ± 0.033	<b>1.5</b>
$\mu$	D	0.033 ± 0.001	<b>0.030</b>
$\alpha$	bohr <sup>3</sup>	0.235 ± 0.061	<b>0.092</b>
$\langle R^2 \rangle$	bohr <sup>2</sup>	<b>0.073 ± 0.002</b>	0.180
$U_0$	eV	<b>0.014 ± 0.001</b>	0.019
$U$	eV	<b>0.019 ± 0.006</b>	<b>0.019</b>
$H$	eV	<b>0.014 ± 0.001</b>	0.017
$G$	eV	<b>0.014 ± 0.000</b>	0.019
$C_v$	cal/mol K	<b>0.033 ± 0.000</b>	0.040

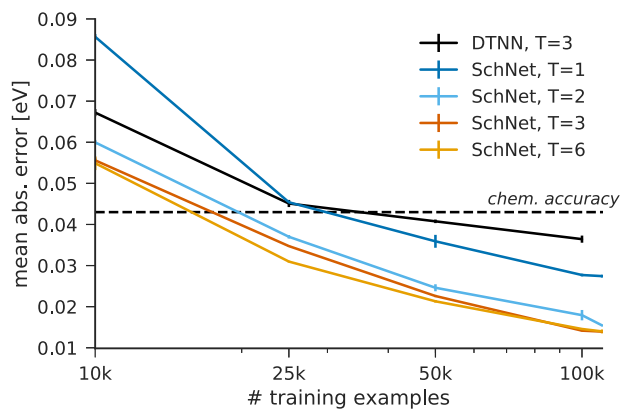


FIG. 3. Mean absolute error (in eV) of energy predictions ( $U_0$ ) on the QM9 dataset<sup>53–55</sup> depending on the number of interaction blocks and reference calculations used for training. For reference, we give the best performing DTNN models ( $T = 3$ ).<sup>28</sup>

to the message-passing neural network *enn-s2s*<sup>29</sup> that use additional bond information beyond atomic positions to learn a molecular representation. The SchNet predictions of the polarizability  $\alpha$  and the electronic spatial extent  $\langle R^2 \rangle$  fall noticeably short in terms of accuracy. This is most likely due to the decomposition of the energy into atomic contributions, which is not appropriate for these properties. In contrast to SchNet, Gilmer *et al.*<sup>29</sup> employed a set2set model variant<sup>56</sup> that obtains a global representation and does not suffer from this issue. However, SchNet reaches or improves over *enn-s2s* in 8 out of 12 properties, where a decomposition into atomic contributions is a good choice. The distributions of the errors of all predicted properties are shown in Appendix A. Extending SchNet with interpretable, property-specific output layers, e.g., for the dipole moment,<sup>57</sup> is subject to future work.

Figure 3 shows learning curves of SchNet for the total energy  $U_0$  with  $T \in \{1, 2, 3, 6\}$  interaction blocks compared to the best performing DTNN models.<sup>28</sup> The best performing DTNN with  $T = 3$  interaction blocks can only outperform the SchNet model with  $T = 1$ . We observe that beyond two interaction blocks the error improves only slightly from 0.015 eV with  $T = 2$  interaction blocks to 0.014 eV for  $T \in \{3, 6\}$  using 110k training examples. When training on fewer examples, the differences become more significant and  $T = 6$ , while having the most parameters, exhibits the lowest errors. Additionally, the model requires much less epochs to converge, e.g., using 110k training examples reduces the required number of epochs from 2400 with  $T = 2$  to less than 750 epochs with  $T = 6$ .

### B. Learning formation energies of materials

We employ SchNet to predict formation energies for bulk crystals using 69 640 structures and reference calculations from the Materials Project (MP) repository.<sup>58,59</sup> It consists of a large variety of bulk crystals with atom type ranging across the whole periodic table up to  $Z = 94$ . Mean absolute errors (MAEs) are listed in Table II. Again, we use  $T = 6$  interaction blocks and atomic representations with  $F = 64$  feature dimension. We set the distance cutoff  $r_{\text{cut}} = 5$  Å and

TABLE II. Mean absolute errors for formation energy predictions in eV/atom on the Materials Project dataset. For SchNet, we give the average over three repetitions as well as standard errors of the mean of the repetitions. Best models in bold.

Model	$N = 3000$	$N = 60\,000$
Ext. Coulomb matrix <sup>5</sup>	0.64	–
Ewald sum matrix <sup>5</sup>	0.49	–
Sine matrix <sup>5</sup>	0.37	–
SchNet ( $T = 6$ )	<b>0.127 ± 0.001</b>	<b>0.035 ± 0.000</b>

discard two examples from the dataset that would include isolated atoms with this setting. Then, the data are randomly split into 60 000 training examples, a validation set of 4500 examples and the remaining data as test set. Even though the MP repository is much more diverse than the QM9 molecule benchmark, SchNet is able to predict formation energies up to a mean absolute error of 0.035 eV/atom. The distribution of the errors is shown in Appendix A. On a smaller subset 3000 training examples, SchNet still achieves an MAE of 0.127 eV/atom improving significantly upon the descriptors proposed by Faber *et al.*<sup>5</sup>

Since the MP dataset contains 89 atom types ranging across the periodic table, we examine the learned atom type embeddings  $\mathbf{x}^0$ . Due to their high dimensionality, we visualize two leading principal components of all sp-atom type embeddings as well as their corresponding group (see Fig. 4). The neural network aims to use the embedding space efficiently such that this 2d projection explains only about 20% of the variance of the embeddings, i.e., since important directions are missing, embeddings might cover each other in the projection while actually being further apart. Still, we already recognize a grouping of elements following the groups of the periodic table. This implies that SchNet has learned that atom types of the same group exhibit similar chemical properties. Within some of the groups, we can even observe an ordering from lighter to heavier elements, e.g., in groups IA and IIA from light elements on the left to heavier ones on the

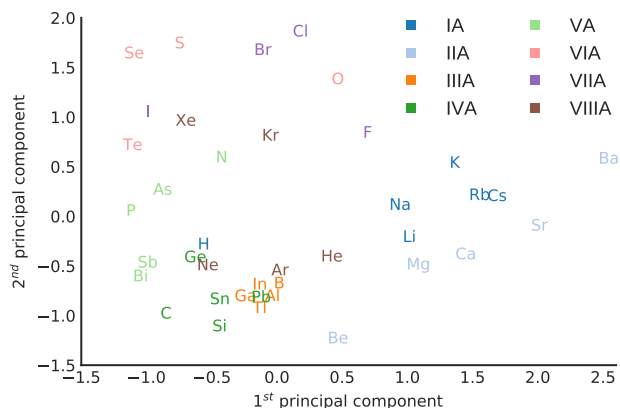


FIG. 4. The two leading principal components of the learned embeddings  $\mathbf{x}^0$  of sp atoms learned by SchNet from the Materials Project dataset. We recognize a structure in the embedding space according to the groups of the periodic table (color-coded) as well as an ordering from lighter to heavier elements within the groups, e.g., in groups IA and IIA from light atoms (left) to heavier atoms (right).

right or, less clear in group VA with a partial ordering N–{As, P}–{Sb, Bi}. Note that this knowledge was not imposed on the machine learning model, but inferred by SchNet from the geometries and formation energy targets of the MP data.

### C. Local chemical potentials

Since the SchNet is a variant of DTNNs, we can visualize the learned representation with a “local chemical potential”  $\Omega_{Z_{\text{probe}}}(\mathbf{r})$  as proposed by Schütt *et al.*:<sup>28</sup> We compute the energy of a virtual atom that acts as a test charge. This can be achieved by adding the probe atom ( $Z_{\text{probe}}, \mathbf{r}_{\text{probe}}$ ) as an input of SchNet. The continuous filter-convolution of the probe atom with the atoms of the system

$$\mathbf{x}_{\text{probe}}^{l+1} = (X^l * W^l)_i = \sum_{i=0}^{n_{\text{atoms}}} \mathbf{x}_i^l \circ W^l(\mathbf{r}_{\text{probe}} - \mathbf{r}_i) \quad (6)$$

ensures that the test charge only senses but does not influence the feature representation. We use Mayavi<sup>60</sup> to visualize the potentials.

Figure 5 shows a comparison of the local potentials of various molecules from QM9 generated by DTNN and SchNet. Both DTNN and SchNet can clearly grasp fundamental chemical concepts such as bond saturation and different degrees of aromaticity. While the general structure of the potential on the surfaces is similar, the SchNet potentials exhibit sharper features and have a more pronounced separation of high-energy and low-energy areas. The overall appearance of the distinguishing molecular features in the “local chemical potentials” is remarkably robust to the underlying neural network architecture, representing the common quantum-mechanical atomic embedding in its molecular environment. It remains to be seen how the “local chemical potentials” inferred by the networks can be correlated with traditional quantum-mechanical observables such as electron density, electrostatic potentials, or electronic orbitals. In addition, such local potentials could aid in the understanding and prediction of chemical reactivity trends.

In the same manner, we show cuts through  $\Omega_C(\mathbf{r})$  for graphite and diamond in Fig. 6. As expected, they resemble the periodic structure of the solid, much like the corresponding filters in Fig. 2. In solids, such local chemical potentials could be used to understand the formation and distribution of defects, such as vacancies and interstitials.

### D. Combined learning of energies and atomic forces

We apply SchNet to the prediction of potential energy surfaces and force fields of the MD17 benchmark set of molecular dynamics trajectories introduced by Chmiela *et al.*<sup>21</sup> MD17 is a collection of eight molecular dynamics simulations for small organic molecules. Tables III and IV list mean absolute errors for energy and force predictions. We trained SchNet on randomly sampled training sets with  $N = 1000$  and  $N = 50\,000$  reference calculations for up to  $2 \times 10^6$  mini-batch gradient steps and additionally used a validation set of 1000 examples for early stopping. The remaining data were used for testing. We also list the performances of gradient domain machine learning (GDML)<sup>21</sup> and DTNN<sup>28</sup> for reference. SchNet was

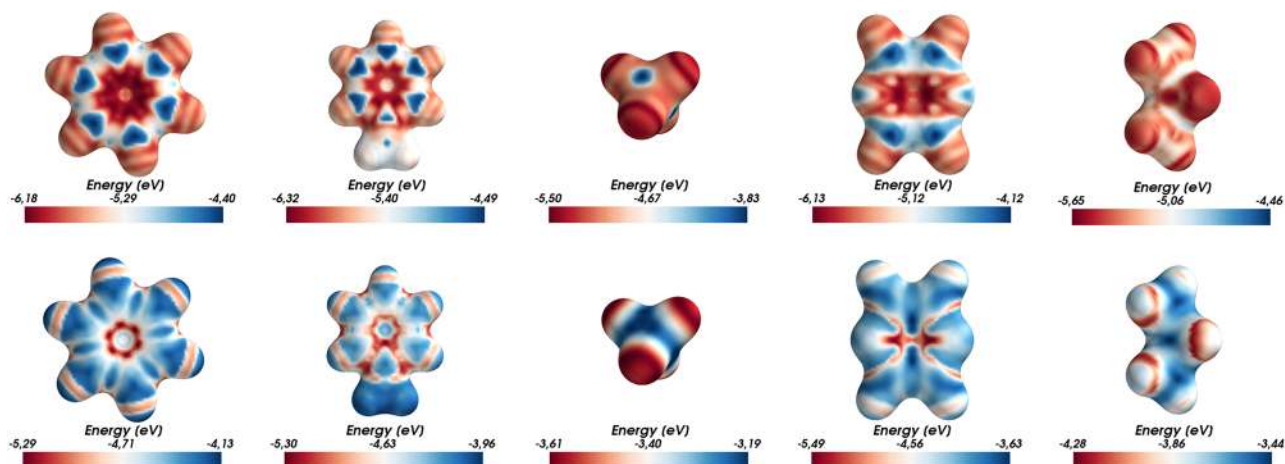


FIG. 5. Local chemical potentials  $\Omega_C(\mathbf{r})$  of DTNN (top) and SchNet (bottom) using a carbon test charge on a  $\sum_i \|\mathbf{r} - \mathbf{r}_i\| = 3.7 \text{ \AA}$  isosurface are shown for benzene, toluene, methane, pyrazine, and propane.

trained with  $T = 3$  interaction blocks and  $F = 64$  feature maps using only energies as well as using the combined loss for energies and forces from Eq. (5) with  $\rho = 0.01$ . This trade-off constitutes a compromise to obtain a single model that performs well on energies and forces for a fair comparison with GDML. Again, we do not use a distance cutoff due to the small molecules and a range up to  $20 \text{ \AA}$  for the Gaussian expansion to cover all distances. In Sec. III E, we will see that even lower errors can be achieved when using two separate SchNet models for energies and forces.

SchNet can take significant advantage of the additional force information, reducing energy, and force errors by 1-2 orders of magnitude compared to energy only training on the small training set. With 50 000 training examples, the improvements are less apparent as the potential energy surface is

already well-sampled at this point. On the small training set, SchNet outperforms GDML on the more flexible molecules malondialdehyde and ethanol, while GDML reaches much lower force errors on the remaining MD trajectories that all include aromatic rings. A possible reason is that GDML defines an order of atoms in the molecule, while the SchNet architecture is inherently invariant to indexing which constitutes a greater advantage in the more flexible molecules.

While GDML is more data-efficient than a neural network, SchNet is scalable to larger datasets. We obtain MAEs of energy and force predictions below  $0.12 \text{ kcal/mol}$  and  $0.33 \text{ kcal}/(\text{mol}\text{\AA})$ , respectively. Remarkably, SchNet performs better while using the combined loss with energies and forces on 1000 reference calculations than training on energies of 50 000 examples.

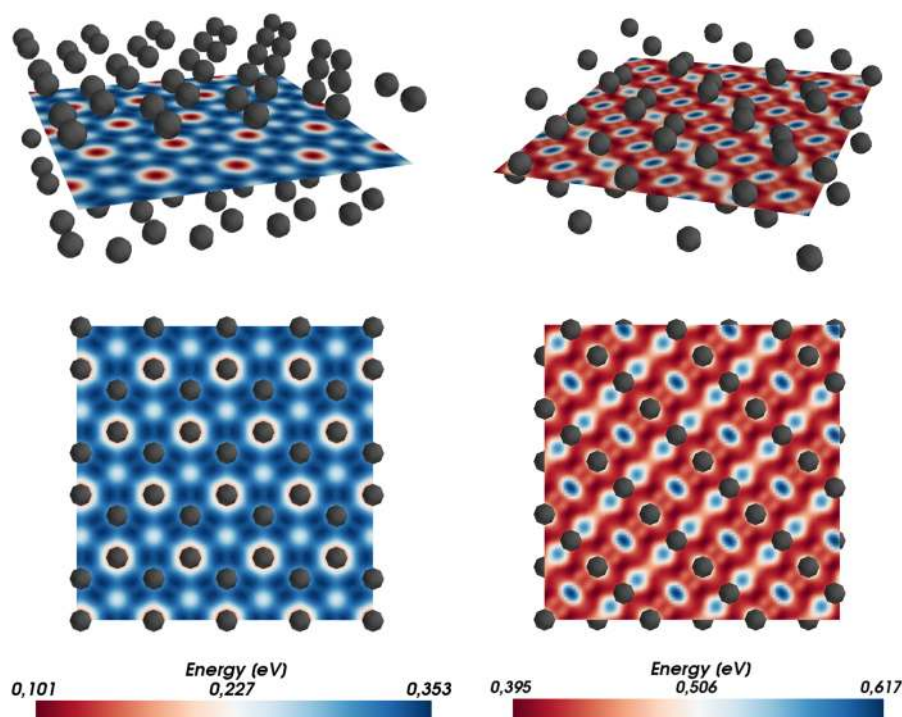


FIG. 6. Cuts through local chemical potentials  $\Omega_C(\mathbf{r})$  of SchNet using a carbon test charge are shown for graphite (left) and diamond (right).

TABLE III. Mean absolute errors for total energies (in kcal/mol). GDML,<sup>21</sup> DTNN,<sup>28</sup> and SchNet<sup>30</sup> test errors for  $N = 1000$  and  $N = 50\,000$  reference calculations of molecular dynamics simulations of small organic molecules are shown. Best results are given in bold.

Trained on	$N = 1000$			$N = 50\,000$		
	GDML Forces	SchNet		DTNN Energy	SchNet	
		Energy	Energy + forces		Energy	Energy + forces
Benzene	<b>0.07</b>	1.19	0.08	<b>0.04</b>	0.08	0.07
Toluene	<b>0.12</b>	2.95	<b>0.12</b>	0.18	0.16	<b>0.09</b>
Malondialdehyde	0.16	2.03	<b>0.13</b>	0.19	0.13	<b>0.08</b>
Salicylic acid	<b>0.12</b>	3.27	0.20	0.41	0.25	<b>0.10</b>
Aspirin	<b>0.27</b>	4.20	0.37	–	0.25	<b>0.12</b>
Ethanol	0.15	0.93	<b>0.08</b>	–	0.07	<b>0.05</b>
Uracil	<b>0.11</b>	2.26	0.14	–	0.13	<b>0.10</b>
Naphthalene	<b>0.12</b>	3.58	0.16	–	0.20	<b>0.11</b>

TABLE IV. Mean absolute errors for atomic forces [in kcal/(mol/Å)]. GDML<sup>21</sup> and SchNet<sup>30</sup> test errors for  $N = 1000$  and  $N = 50\,000$  reference calculations of molecular dynamics simulations of small organic molecules are shown. Best results are given in bold.

Trained on	$N = 1000$			$N = 50\,000$	
	GDML Forces	SchNet		SchNet	
		Energy	Energy + forces	Energy	Energy + forces
Benzene	<b>0.23</b>	14.12	0.31	1.23	<b>0.17</b>
Toluene	<b>0.24</b>	22.31	0.57	1.79	<b>0.09</b>
Malondialdehyde	0.80	20.41	<b>0.66</b>	1.51	<b>0.08</b>
Salicylic acid	<b>0.28</b>	23.21	0.85	3.72	<b>0.19</b>
Aspirin	<b>0.99</b>	23.54	1.35	7.36	<b>0.33</b>
Ethanol	0.79	6.56	<b>0.39</b>	0.76	<b>0.05</b>
Uracil	<b>0.24</b>	20.08	0.56	3.28	<b>0.11</b>
Naphthalene	<b>0.23</b>	25.36	0.58	2.58	<b>0.11</b>

### E. Application to molecular dynamics of $C_{20}$ -fullerene

After demonstrating the accuracy of SchNet on the MD17 benchmark set, we perform a study of a ML-driven MD simulation of  $C_{20}$ -fullerene. This middle-sized molecule has a complex PES that requires to be described with accuracy to reproduce vibrational normal modes and their degeneracies. Here, we use SchNet to perform an analysis of some basic properties of the PES of  $C_{20}$  when introducing nuclear quantum effects (NQE). The reference data were generated by running classical MD at 500 K using DFT at the generalized gradient approximation (GGA) level of theory with the Perdew-Burke-Ernzerhof (PBE)<sup>39</sup> exchange-correlation functional and the Tkatchenko-Scheffler (TS) method<sup>40</sup> to account for van der Waals interactions. Further details about the simulations can be found in [Appendix B](#).

By training SchNet on DFT data at the PBE+vdW<sup>TS</sup> level, we reduce the computation time per single point by three orders of magnitude from 11 s using 32 CPU cores to 10 ms using one NVIDIA GTX1080. This allows us to perform long MD simulations with DFT accuracy at low computational cost, making this kind of study feasible.

In order to obtain accurate energy and force predictions, we first perform an extensive model selection on the given reference data. We use 20k  $C_{20}$  reference calculations as the training set, 4.5k examples for early stopping, and report the test error on the remaining data. Table [V](#) lists the results for

TABLE V. Mean absolute errors for energy and force predictions of  $C_{20}$ -fullerene in kcal/mol and kcal/(mol/Å), respectively. We compare SchNet models with varying numbers of interaction blocks  $T$ , feature dimensions  $F$ , and energy-force tradeoff  $\rho$ . For force-only training ( $\rho = 0$ ), the integration constant is fitted separately. Best models in bold.

$T$	$F$	$\rho$	Energy	Forces
3	64	0.010	0.228	0.401
6	64	0.010	0.202	0.217
3	128	0.010	0.188	0.197
6	128	0.010	<b>0.1002</b>	<b>0.120</b>
6	128	0.100	<b>0.027</b>	0.171
6	128	0.010	0.100	0.120
6	128	0.001	0.238	0.061
6	128	0.000	0.260	<b>0.058</b>



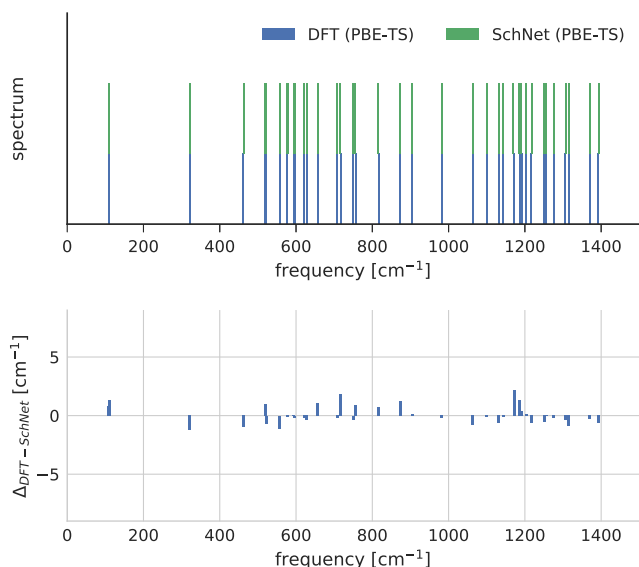


FIG. 7. Normal mode analysis of the fullerene  $C_{20}$  dynamics comparing SchNet and DFT results.

various settings of number of interaction blocks  $T$ , number of feature dimensions  $F$  of the atomic representations, and the energy-force trade-off  $\rho$  of the combined loss function. First, we select the best hyper-parameters  $T$ ,  $F$  of the model given the trade-off  $\rho = 0.01$  that we established to be a good compromise on MD17 (see the upper part of Table V). We find that the configuration of  $T = 6$  and  $F = 128$  works best for energies as well as forces. Given the selected model, we next validate the best choice for the trade-off  $\rho$ . Here we find that the best choices for energy and forces vastly diverge: While we established before that energy predictions benefit from force information (see Table III), we achieve the best force predictions for  $C_{20}$ -fullerene when neglecting the energies. We still benefit from using the derivative of an energy model as force

model since this still guarantees an energy-conserving force field.<sup>21</sup>

For energy predictions, we obtain the best results when using a larger  $\rho = 0.1$  as this puts more emphasis on the energy loss. Here, we select the force-only model as force field to drive our MD simulation since we are interested in the mechanical properties of the  $C_{20}$  fullerene. Figure 7 shows a comparison of the normal modes obtained from DFT and our model. In the bottom panel, we show the accuracy of SchNet with the largest error being  $\sim 1\%$  of the DFT reference frequencies. Given these results and the accuracy reported in Table V, we obtained a model that successfully reconstructs the PES and its symmetries.<sup>61</sup>

In addition, in Fig. 8, we present an analysis of the nearest neighbor (1nn), diameter and radial distribution functions at 300 K for classical MD (blue) and PIMD (green) simulations that include nuclear quantum effects. See Appendix B for further details on the simulation. From Fig. 8 (and Fig. 11), it looks like nuclear delocalization does not play a significant role in the peaks of the pair distribution function  $h(r)$  for  $C_{20}$  at room temperature. The nuclear quantum effects increase the 1nn distances by less than 0.5% but the delocalization of the bond lengths is considerable. This result agrees with previously reported PIMD simulations of graphene.<sup>62</sup> However, here we have non-symmetric distributions due to the finite size of  $C_{20}$ .

Overall, with SchNet we could carry out 1.25 ns of PIMD, reducing the runtime compared to DFT by 3-4 orders of magnitude: from about 7 years to less than 7 h with much less computational resources. Such long time MD simulations are required for detailed studies on mechanical and thermodynamical properties as a function of the temperature, especially in the low temperature regime, where the nuclear quantum effects become extremely important. Clearly, this application evinces the need for fast and accurate machine learning model such as SchNet to explore the different nature of chemical

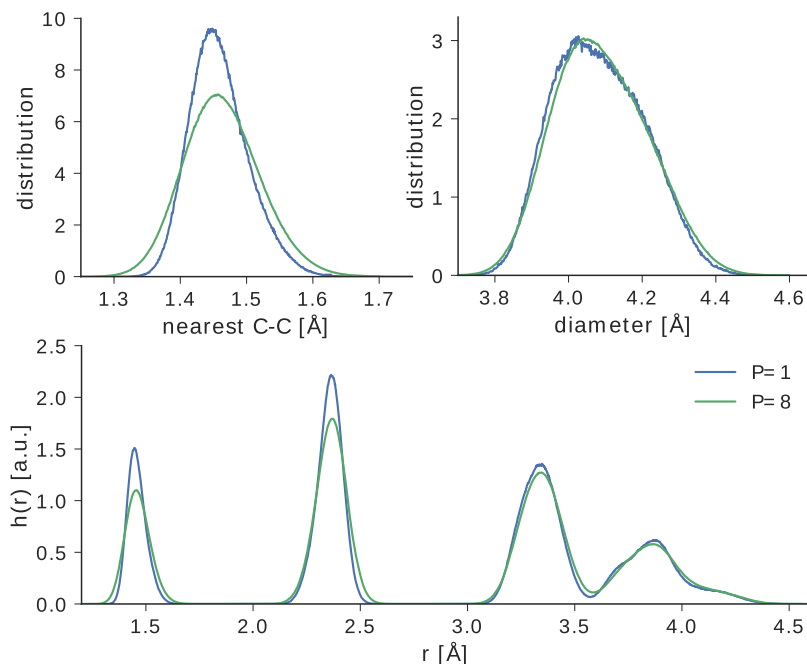


FIG. 8. Analysis of the fullerene  $C_{20}$  dynamics at 300 K using SchNet@DFT. Distribution functions for nearest neighbours, diameter of the fullerene, and the atomic-pair distribution function using classical MD (blue) and PIMD (green) with 8 beads.

interactions and quantum behavior to better understand molecules and materials.

#### IV. CONCLUSIONS

Instead of having to painstakingly design mechanistic force fields or machine learning descriptors, deep learning allows us to learn a representation from first principles that adapts to the task and scale at hand, from property prediction across chemical compound space to force fields in the configurational space of single molecules. The design challenge here has been shifted to modeling quantum interactions by choosing a suitable neural network architecture. This gives rise to the possibility to encode known quantum-chemical constraints and symmetries within the model without losing the flexibility of a neural network. This is crucial in order to be able to accurately represent, e.g., the full potential-energy surface and, in particular, its anharmonic behavior.

We have presented the deep learning architecture SchNet which can be applied to a variety of applications ranging from the prediction of chemical properties for diverse datasets of molecules and materials to highly accurate predictions of potential energy surfaces and energy-conserving force fields. As a variant of DTNNs, SchNet follows rotational, translational, and permutational invariances by design and, beyond that, is able to directly model periodic boundary conditions. Not only does SchNet yield fast and accurate predictions, but it also allows us to examine the learned representation using local chemical potentials.<sup>28</sup> Beyond that, we have analyzed the atomic embeddings learned by SchNet and found that fundamental chemical knowledge had been recovered purely

from a dataset of bulk crystals and formation energies. Most importantly, we have performed an exemplary path-integral molecular dynamics study on the fullerene  $C_{20}$  at the PBE+vdW<sup>TS</sup> level of theory that would not have been computationally feasible with common DFT approaches. These encouraging results will guide future work such as studies of larger molecules and periodic systems as well as further developments toward interpretable deep learning architectures to assist chemistry research.

#### ACKNOWLEDGMENTS

This work was supported by the Federal Ministry of Education and Research (BMBF) for the Berlin Big Data Center BBDC (No. 01IS14013A). Additional support was provided by the DFG (No. MU 987/20-1), from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie Grant Agreement No. 657679, the BK21 program funded by the Korean National Research Foundation Grant (No. 2012-005741) and the Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (No. 2017-0-00451). A.T. acknowledges support from the European Research Council (ERC-CoG grant BeStMo).

#### APPENDIX A: ERROR DISTRIBUTIONS

In Figs. 9 and 10, we show histograms of the predicted properties of the QM9 and Materials Project dataset, respectively. The histograms include all test errors made across all three repetitions.

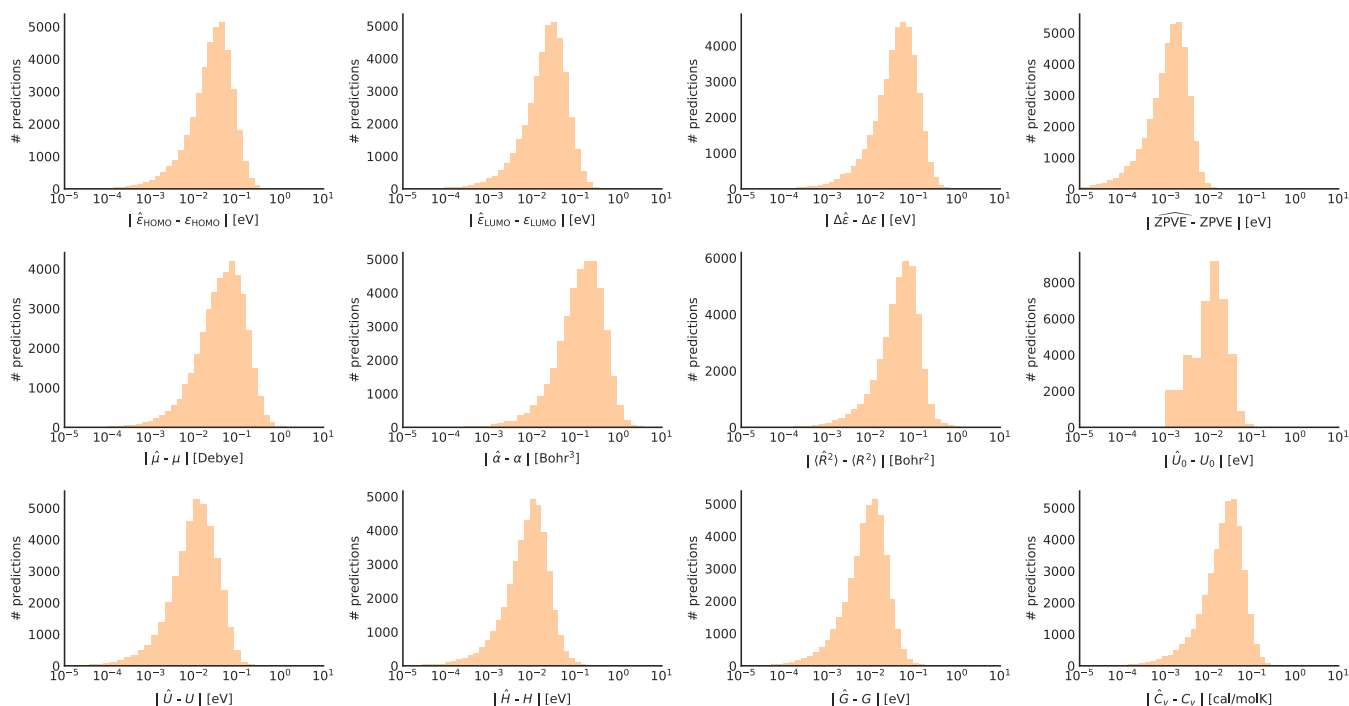


FIG. 9. Histograms of absolute errors for all predicted properties of QM9. The histograms are plotted on a logarithmic scale to visualize the tails of the distribution.

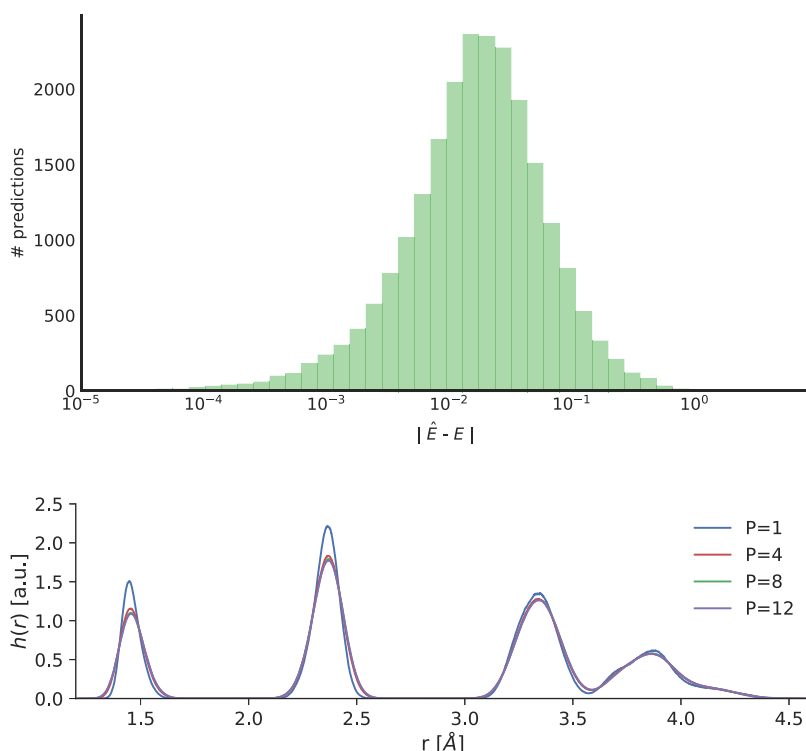


FIG. 10. Histogram of absolute errors for the predictions of formation energies/atom for the Materials Project dataset. The histogram is plotted on a logarithmic scale to visualize the tails of the distribution.

FIG. 11. Histograms of absolute errors for all predicted properties of QM9. The histograms are plotted on a logarithmic scale to visualize the tails of the distribution.

## APPENDIX B: MD SIMULATION DETAILS

The reference data for  $C_{20}$  were generated using classical molecular dynamics in the NVT ensemble at 500 K using the Nose-Hoover thermostat with a time step of 1 fs. The forces and energies were computed using DFT with the generalized gradient approximation (GGA) level of theory with the non-empirical exchange-correlation functional of Perdew-Burke-Ernzerhof (PBE)<sup>39</sup> and the Tkatchenko-Scheffler (TS) method<sup>40</sup> to account for ubiquitous van der Waals interactions. The calculations were done using all-electrons with a light basis set implemented in the FHI-aims code.<sup>63</sup>

The quantum nuclear effects are introduced using path-integral molecular dynamics (PIMD) via Feynman's path integral formalism. The PIMD simulations were done using the SchNet model implementation in the i-PI code.<sup>64</sup> The integration time step was set to 0.5 fs to ensure energy conservation along the MD using the NVT ensemble with a stochastic path integral Langevin equation (PILE) thermostat.<sup>65</sup> In PIMD, the treatment of NQE is controlled by the number of beads,  $P$ . In our example for  $C_{20}$  fullerene, we can see that at room temperature using 8 beads gives an already converged radial distribution function  $h(r)$  as shown in Fig. 11.

<sup>1</sup>M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, *Phys. Rev. Lett.* **108**, 058301 (2012).

<sup>2</sup>G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, *New J. Phys.* **15**, 095003 (2013).

<sup>3</sup>K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, and K.-R. Müller, *J. Chem. Theory Comput.* **9**, 3404 (2013).

<sup>4</sup>K. T. Schütt, H. Glawe, F. Brockherde, A. Sanna, K.-R. Müller, and E. Gross, *Phys. Rev. B* **89**, 205118 (2014).

<sup>5</sup>F. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, *Int. J. Quantum Chem.* **115**, 1094 (2015).

<sup>6</sup>R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, *J. Chem. Theory Comput.* **11**, 2087 (2015).

<sup>7</sup>K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O. A. von Lilienfeld, K.-R. Müller, and A. Tkatchenko, *J. Phys. Chem. Lett.* **6**, 2326 (2015).

<sup>8</sup>F. A. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, *Phys. Rev. Lett.* **117**, 135502 (2016).

<sup>9</sup>M. Hirn, S. Mallat, and N. Poilvert, *Multiscale Model. Simul.* **15**, 827 (2017).

<sup>10</sup>F. A. Faber, L. Hutchison, B. Huang, J. Gilmer, S. S. Schoenholz, G. E. Dahl, O. Vinyals, S. Kearnes, P. F. Riley, and O. A. von Lilienfeld, *J. Chem. Theory Comput.* **13**(11), 5255–5264 (2017).

<sup>11</sup>H. Huo and M. Rupp, preprint [arXiv:1704.06439](https://arxiv.org/abs/1704.06439) (2017).

<sup>12</sup>M. Eickenberg, G. Exarchakis, M. Hirn, and S. Mallat, *Advances in Neural Information Processing Systems 30* (Curran Associates, Inc., 2017), pp. 6522–6531.

<sup>13</sup>O. Isayev, C. Oses, C. Toher, E. Gossett, S. Curtarolo, and A. Tropsha, *Nat. Commun.* **8**, 15679 (2017).

<sup>14</sup>K. Ryczko, K. Mills, I. Luchak, C. Homenick, and I. Tamblin, preprint [arXiv:1706.09496](https://arxiv.org/abs/1706.09496) (2017).

<sup>15</sup>I. Luchak, K. Mills, K. Ryczko, A. Domurad, and I. Tamblin, preprint [arXiv:1708.06686](https://arxiv.org/abs/1708.06686) (2017).

<sup>16</sup>J. Behler and M. Parrinello, *Phys. Rev. Lett.* **98**, 146401 (2007).

<sup>17</sup>J. Behler, *J. Chem. Phys.* **134**, 074106 (2011).

<sup>18</sup>A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, *Phys. Rev. Lett.* **104**, 136403 (2010).

<sup>19</sup>A. P. Bartók, R. Kondor, and G. Csányi, *Phys. Rev. B* **87**, 184115 (2013).

<sup>20</sup>A. V. Shapeev, *Multiscale Model. Simul.* **14**, 1153 (2016).

<sup>21</sup>S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K.-R. Müller, *Sci. Adv.* **3**, e1603015 (2017).

<sup>22</sup>F. Brockherde, L. Voigt, L. Li, M. E. Tuckerman, K. Burke, and K.-R. Müller, *Nat. Commun.* **8**, 872 (2017).

<sup>23</sup>J. S. Smith, O. Isayev, and A. E. Roitberg, *Chem. Sci.* **8**, 3192 (2017).

<sup>24</sup>E. V. Podryabinkin and A. V. Shapeev, *Comput. Mater. Sci.* **140**, 171 (2017).

<sup>25</sup>P. Rowe, G. Csányi, D. Alfè, and A. Michaelides, *Phys. Rev. B* **97**(5), 054303 (2018).

<sup>26</sup>D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams, in *Conference on Neural Information Processing Systems*, edited by C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (Curran Associates, Inc., 2015), pp. 2224–2232.

<sup>27</sup>S. Kearnes, K. McCloskey, M. Berndl, V. Pande, and P. F. Riley, *J. Comput.-Aided Mol. Des.* **30**, 595 (2016).

- <sup>28</sup>K. T. Schütt, F. Arbabzadah, S. Chmiela, K.-R. Müller, and A. Tkatchenko, *Nat. Commun.* **8**, 13890 (2017).
- <sup>29</sup>J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, in *Proceedings of the 34th International Conference on Machine Learning* (PMLR, 2017), pp. 1263–1272.
- <sup>30</sup>K. T. Schütt, P.-J. Kindermans, H. E. Sauceda, S. Chmiela, A. Tkatchenko, and K.-R. Müller, in *Advances in Neural Information Processing Systems 30* (Curran Associates, Inc., 2017), pp. 992–1002.
- <sup>31</sup>D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K.-R. Müller, *J. Mach. Learn. Res.* **11**, 1803–1831 (2010).
- <sup>32</sup>K. Simonyan, A. Vedaldi, and A. Zisserman, eprint [arXiv:1312.6034](https://arxiv.org/abs/1312.6034) (2013).
- <sup>33</sup>S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, *PLoS One* **10**, e0130140 (2015).
- <sup>34</sup>L. M. Zintgraf, T. S. Cohen, T. Adel, and M. Welling, in *International Conference on Learning Representations*, 2017.
- <sup>35</sup>G. Montavon, S. Lapuschkin, A. Binder, W. Samek, and K.-R. Müller, *Pattern Recognit.* **65**, 211 (2017).
- <sup>36</sup>P.-J. Kindermans, K. T. Schütt, M. Alber, K.-R. Müller, D. Erhan, B. Kim, and S. Dähne, eprint [arXiv:1705.05598](https://arxiv.org/abs/1705.05598) (2017).
- <sup>37</sup>G. Montavon, W. Samek, and K.-R. Müller, *Digital Signal Process.* **73**, 1 (2018).
- <sup>38</sup>K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, in *International Conference on Machine Learning* (PMLR, 2015), pp. 2048–2057.
- <sup>39</sup>J. P. Perdew, K. Burke, and M. Ernzerhof, *Phys. Rev. Lett.* **77**, 3865 (1996).
- <sup>40</sup>A. Tkatchenko and M. Scheffler, *Phys. Rev. Lett.* **102**, 073005 (2009).
- <sup>41</sup>I. Poltavsky and A. Tkatchenko, *Chem. Sci.* **7**, 1368 (2016).
- <sup>42</sup>G. W. Taylor and G. E. Hinton, in *Proceedings of the 26th Annual International Conference on Machine Learning ICML 09* (ACM, 2009), Vol. 49, p. 1.
- <sup>43</sup>D. Yu, L. Deng, and F. Seide, *IEEE Trans. Audio, Speech, Lang. Process.* **21**, 388 (2013).
- <sup>44</sup>R. Socher, A. Perelygin, J. Y. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts, in *Conference on Empirical Methods in Natural Language Processing* (ACL, 2013), Vol. 1631, p. 1642.
- <sup>45</sup>X. Jia, B. De Brabandere, T. Tuytelaars, and L. V. Gool, in *Advances in Neural Information Processing Systems 29*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Inc., 2016), pp. 667–675.
- <sup>46</sup>Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, *Neural Comput.* **1**, 541 (1989).
- <sup>47</sup>A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Advances in Neural Information Processing Systems* (Curran Associates, Inc., 2012), pp. 1097–1105.
- <sup>48</sup>A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, “WaveNet: A Generative Model for Raw Audio,” [arXiv:1609.03499](https://arxiv.org/abs/1609.03499) (2016).
- <sup>49</sup>F. Chollet, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (IEEE, 2017), pp. 1251–1258.
- <sup>50</sup>K. He, X. Zhang, S. Ren, and J. Sun, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (IEEE, 2016), pp. 770–778.
- <sup>51</sup>A. Pukrittayakamee, M. Malshe, M. Hagan, L. Raff, R. Narulkar, S. Bukkapatnum, and R. Komanduri, *J. Chem. Phys.* **130**, 134101 (2009).
- <sup>52</sup>D. P. Kingma and J. Ba, in *International Conference on Learning Representations*, 2015.
- <sup>53</sup>R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, *Sci. Data* **1**, 140022 (2014).
- <sup>54</sup>L. C. Blum and J.-L. Reymond, *J. Am. Chem. Soc.* **131**, 8732 (2009).
- <sup>55</sup>J.-L. Reymond, *Acc. Chem. Res.* **48**, 722 (2015).
- <sup>56</sup>O. Vinyals, S. Bengio, and M. Kudlur, eprint [arXiv:1511.06391](https://arxiv.org/abs/1511.06391) (2015).
- <sup>57</sup>M. Gastegger, J. Behler, and P. Marquetand, *Chem. Sci.* **8**(10), 6924–6935 (2017).
- <sup>58</sup>A. Jain, S. P. Ong, G. Hautier, W. Chen, W. D. Richards, S. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K. A. Persson, *APL Mater.* **1**, 011002 (2013).
- <sup>59</sup>S. P. Ong, W. D. Richards, A. Jain, G. Hautier, M. Kocher, S. Cholia, D. Gunter, V. L. Chevrier, K. A. Persson, and G. Ceder, *Comput. Mater. Sci.* **68**, 314 (2013).
- <sup>60</sup>P. Ramachandran and G. Varoquaux, *Comput. Sci. Eng.* **13**, 40 (2011).
- <sup>61</sup>Code and trained models are available at: <https://github.com/atomistic-machine-learning/SchNet>.
- <sup>62</sup>I. Poltavsky, R. A. DiStasio, Jr., and A. Tkatchenko, *J. Chem. Phys.* **148**, 102325 (2018).
- <sup>63</sup>V. Blum, R. Gehrke, F. Hanke, P. Havu, V. Havu, X. Ren, K. Reuter, and M. Scheffler, *Comput. Phys. Commun.* **180**, 2175 (2009).
- <sup>64</sup>M. Ceriotti, J. More, and D. E. Manolopoulos, *Comput. Phys. Commun.* **185**, 1019 (2014).
- <sup>65</sup>M. Ceriotti, M. Parrinello, T. E. Markland, and D. E. Manolopoulos, *J. Chem. Phys.* **133**, 124104 (2010).