



TECHNISCHE
UNIVERSITÄT
WIEN

MASTER'S THESIS

Prediction of Born effective charges via Neural Network Potentials: Design of expressive electric field descriptors

carried out for the purpose of obtaining the degree of Master of Science, submitted at
TU Wien, Institute of Materials Chemistry, by

Erwin Rait, BSc.

Mat.Nr.: 11901979

under the supervision of

Univ. Prof. Georg K. H. Madsen, PhD

E165 Institut für Materialchemie

Technische Universität Wien

Univ. Prof. Satoshi Watanabe, PhD

Department of Materials Engineering

University of Tokyo

Vienna, September 2025

Abstract

Understanding ionic behavior is essential for developing electronic devices that rely on ion transport. Although machine-learned force fields have become the standard for large-scale materials simulations, they do not explicitly account for electrostatic interactions. As a result, they are not well-suited for studying systems where ions interact with an external electric field. The Born effective charge (BEC) - a 3×3 tensor defined as the derivative of forces with respect to an external electric field - offers a practical way to compute forces arising from electrostatic interactions. However, calculating BECs using ab-initio methods is computationally expensive, creating a need for more efficient approaches. Since BECs, similar to forces, can be expressed as derivatives of the potential energy, it is possible to obtain them directly by differentiating the output of a neural network potential (NNP) within a unified model [16]. This approach not only improves computational efficiency but also ensures that the resulting BECs satisfy exact physical constraints. In this work, I adapt NeuralIL - a NNP that uses automatic differentiation to compute forces [38] - and demonstrate that, with carefully designed descriptors, it is possible to predict Born effective charges with sufficient accuracy to perform molecular dynamics simulations under external electric fields.

要旨

イオン輸送に依存する新たな電子デバイスを開発するためには、イオン挙動の正しい理解が必要である。大規模な材料シミュレーションでは、機械学習による力場が標準化されつつあるが、その多くは静電相互作用を明示的に考慮しておらず、イオンが外部電場と相互作用する場合には適さない。よって、電場印加時の静電相互作用を計算するため、ボルン有効電荷という、外部電場に対する力の導関数として定義される 3×3 テンソルによる計算手法が用いられる。しかし、第一原理計算によるボルン有効電荷の算出は計算コストが非常に高く、大規模スケールの分子動力学計算は困難であるため、より効率的な手法が求められている。一方、ボルン有効電荷は、力と同様に位置エネルギーの導関数として表すことができるため、ニューラルネットワークポテンシャルの出力を微分することで、統合モデル [16] で直接計算することができる。これにより、計算効率を向上できるうえ、得られるボルン有効電荷は厳密な物理的拘束条件を満たす。本研究では、自動微分により力を求める NeuralIL [38] というニューラルネットワークポテンシャルを拡張し、電場情報を含めた電気化学的記述子を設計した。この結果、電場印加時の分子動力学計算を実現する上で、ボルン有効電荷を十分に高い精度で予測することができた。

Zusammenfassung

Das Verständnis des Ionenverhaltens ist entscheidend für die Entwicklung elektronischer Geräte, die auf Ionentransport basieren. Obwohl maschinell gelernte Kraftfelder mittlerweile zum Standard für Materialsimulationen auf großen Skalen geworden sind, werden elektrostatische Wechselwirkungen nicht explizit inkludiert. Daher erweisen sie sich als ungeeignet für die Untersuchung von Systemen, in denen Ionen mit einem externen elektrischen Feld interagieren. Die effektive Born Ladung (BEC) - ein 3×3 -Tensor, definiert als die Ableitung der Kräfte bezüglich eines externen elektrischen Feldes - bietet eine direkte Methode zur Berechnung der durch elektrostatische Wechselwirkungen entstehenden Kräfte. Die Berechnung von BECs mit ab-initio-Methoden ist jedoch rechenintensiv, was den Bedarf an effizienteren Methoden schafft. Da sich BECs, ähnlich wie Kräfte, als Ableitung der potentiellen Energie darstellen lassen, ist es möglich, sie direkt durch Differenzieren der Ausgabe eines neuronalen Netzwerkpotentials (NNP) innerhalb eines einheitlichen Modells zu bestimmen [16]. Dieser Ansatz verbessert nicht nur die Recheneffizienz, sondern stellt auch sicher, dass die resultierenden BECs exakte physikalische Randbedingungen erfüllen. In dieser Arbeit adaptiere ich NeuralIL - ein NNP, das automatische Differenzierung zur Berechnung von Kräften verwendet [38] - und zeige, dass es mit sorgfältig entworfenen Deskriptoren möglich ist, effektive Born Ladungen mit ausreichender Genauigkeit vorherzusagen, um Molekulardynamiksimulationen unter externen elektrischen Feldern durchzuführen.

Contents

1	Introduction	1
2	Neural Network Force Fields	3
2.1	Artificial Neural Networks	3
2.2	Representation of the PES	5
2.3	NeuralIL	8
2.4	Graph convolutional neural networks	10
3	Born effective charges	13
3.1	Computation	14
3.2	Prediction using ML	14
4	Methodology	17
4.1	Training dataset	17
4.2	Born effective charge prediction using forces	19
4.3	Born effective charge prediction in unified framework	21
5	Results	29
5.1	Born effective charge prediction using forces	29
5.2	Born effective charge prediction in unified framework	31
6	Conclusion and Outlook	43
	Bibliography	45
	List of Abbreviations	51
A	Methodology	53
B	Results	57

List of Figures

2.1	Representation of the overall NEURALIL model, including generation of spherical Bessel descriptors, computation of embedding vectors, final sum over atomic contributions to E_{pot} and computation of forces. Taken from [38].	9
3.1	Schematic representation of uniform differentiable learning of electric response from [16]. The E_{pot} output of a neural network potential (NNP) is differentiated with respect to it's inputs to obtain various properties.	15
4.1	Histogram showing the distribution of Born effective charges (BECs) calculated in Vienna ab-initio simulation package (VASP) for a dataset consisting of 224 water dimers.	18
4.2	One high symmetry structure from the water dimer dataset. All atoms lie inside a plane and a C_2 -axis goes through the oxygen atoms.	18
4.3	Histogram showing the distribution of BECs calculated in VASP for ZrO_2 DEFECT	19
4.4	(a) Example structure from ZrO_2 NODEFECT with cubic symmetry. (b) Orthorhombic example structure from the Li_3PO_4 dataset. Phosphorus is represented in orange, lithium in purple.	20
4.5	Incorporation of $\vec{\mathcal{E}}$ as an input of NEURALIL. Modified from [38].	22
4.6	Comparison of the effective radial basis used in RADIAL2 on the left and RADIAL on the right.	23
4.7	Derivation of ANGULAR. One of the neighbors, k , in the double sum (a) is replaced by a pseudo atom at a distance $\vec{\mathcal{E}}$ from i (b).	24
5.1	Fitting of forces $\Delta\vec{F}_{\text{ext}}$ induced by the electric field $\vec{\mathcal{E}}_z$ using NEURALIL. . .	29
5.2	Fitting of forces $\Delta\vec{F}_{\text{ext}}$ induced by electric fields $\vec{\mathcal{E}}_x$ and $\vec{\mathcal{E}}_y$ using NEURALIL.	30
5.3	Induced forces $\Delta\vec{F}_{\text{ext}}$ shown schematically for (a) $\vec{\mathcal{E}}_z$, (b) $\vec{\mathcal{E}}_y$ and (c) $\vec{\mathcal{E}}_x$. . .	31
5.4	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the water dimer dataset using the unified differentiable learning of electric response framework with the IDENTITY descriptors.	32
5.5	Prediction of BECs in the water dimer dataset using the unified differentiable learning of electric response framework with the IDENTITY descriptors. . . .	33

5.6	Depiction of how ideally the descriptors used in the NNP should behave under rotation of both positions $\vec{R}_i$ and electric field vector $\vec{\mathcal{E}}$	34
5.7	Depiction of how ideally the descriptors used in the NNP should behave under rotation of positions $\vec{R}_i$	34
5.8	Prediction of BECs in the water dimer dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	36
5.9	Prediction of BECs in the ZrO ₂ NoDEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	36
5.10	Final validation set performance of NEURALIL models on the ZrO ₂ DEFECT dataset. Visual representation of data in table 5.2. Models marked with * are trained with a log cosh parameter of $10e$ for the BEC and $\lambda_{\text{BEC}} = 20$	38
5.11	Prediction of BECs in the ZrO ₂ DEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	38
5.12	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO ₂ DEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	39
5.13	Prediction of BECs in the ZrO ₂ DEFECT dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.	40
5.14	Prediction of BECs in the ZrO ₂ DEFECT dataset using the unified differentiable learning of electric response framework with the ELEMENT descriptors. Trained with a BEC log cosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$	40
5.15	Prediction of BECs in the Li ₃ PO ₄ dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.	41
5.16	Final validation set performance of NEURALIL models on the Li ₃ PO ₄ dataset. For each model, 4 repetitions were carried out. Models marked with * are trained with a log cosh parameter of $10e$ for the BEC and $\lambda_{\text{BEC}} = 20$	42
5.17	Prediction of BECs in the Li ₃ PO ₄ dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors. Trained with a BEC log cosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$	42
A.1	Histogram showing the distribution of BECs calculated in VASP for ZrO ₂ NoDEFECT	53
A.2	Histogram showing the distribution of BECs calculated in VASP for a dataset of pristine Li ₃ PO ₄ generated with <i>ab initio</i> molecular dynamics (AIMD) at 2000 K.	54
A.3	One tetragonal structure (mp-754403) from ZrO ₂ DEFECT.	55
A.4	(a) One tetragonal structure (mp-2574) and (b) one monoclinic structure (mp-2858) from ZrO ₂ DEFECT.	55
B.1	Fitting of forces $\Delta\vec{F}_{\text{ext}}$ induced by the electric field $\vec{\mathcal{E}}_z$ using MACE.	57
B.2	Fitting of forces $\Delta\vec{F}_{\text{ext}}$ induced by electric fields $\vec{\mathcal{E}}_x$ and $\vec{\mathcal{E}}_y$ using MACE.	58

B.3	Prediction of polarizations $\vec{P}$ and polarizabilities α in the water dimer dataset using the unified differentiable learning of electric response framework with the IDENTITY descriptors.	58
B.4	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the water dimer dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	59
B.5	Prediction of polarizations $\vec{P}$ and polarizabilities α in the water dimer dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	59
B.6	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO_2 NODE- FECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.	60
B.7	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO_2 DEFECT dataset using the unified differentiable learning of electric response frame- work with the EBESSEL descriptors.	60
B.8	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO_2 DEFECT dataset using the unified differentiable learning of electric response frame- work with the ELEMENT descriptors. Trained with a BEC log cosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$	61
B.9	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the Li_3PO_4 dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.	61
B.10	Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the Li_3PO_4 dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors. Trained with a BEC log cosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$	62

Chapter 1

Introduction

When developing new electronic and energy-related devices, such as batteries and atomic switches, it is crucial to understand how atoms, charges, and external electric fields interact during device operation. While experiments provide valuable insights into these systems, a detailed understanding at the atomic scale remains largely elusive.

Computational materials science offers a powerful approach to explore materials at this level of resolution, by performing molecular dynamics (MD) simulations one can study the time evolution of a system. However, accurate forces are needed at each time step. One option is to use *ab initio* methods to calculate the forces, but the high computational resources needed, limit AIMD simulations to small systems and short timescales. A more efficient alternative of obtaining forces is to use classical force field, such as OPLS-AA [26] or REAXFF [44], which are considered state of the art force fields for MD. These models are constructed by parametrizing functional forms that are usually physically motivated. Even though they are very fast to evaluate, they lack accuracy and transferability needed to study subtle or complex phenomena.

In recent years, machine learning force field (MLFF) have emerged to bridge the gap between highly accurate but poorly scalable *ab initio* methods and very low computational cost but not very accurate classical force fields [11]. The problem of calculating forces is viewed as a simple regression problem, relying on no arbitrary assumptions on the functional forms. MLFFs have enabled simulations of unprecedented accuracy and scale.

Despite this progress, most machine learning (ML) models do not account for electrostatic interactions, making them unsuitable to study systems where these interactions play a major role. Particularly, in systems with an applied electric fields, it is necessary to know the forces induced by the electric field. The physical quantity that describes this interaction is the BEC, thus making it an ideal quantity in the study of such systems [48]. However, the traditional *ab-initio* method for calculating BECs is computationally demanding, motivating the search for alternative methods that provide accurate BECs

at significantly reduced computational cost. Recently, considerable effort has gone into using, similar to MLFFs, ML to predict BECs [33, 46].

In this thesis, NEURALIL [14, 38] a NNP, that uses local invariant descriptors, is adapted to predict BEC alongside forces, potential energy and other dielectric properties within a unified model [16]. Great care is taken in including the electric field information inside newly designed descriptors, ensuring that both symmetry is preserved and the descriptors are expressive. Multiple descriptors are tested and compared, with the ultimate goal of enabling large scale MD simulations under external electric fields.

Chapter 2

Neural Network Force Fields

The traditional way of studying chemical systems is the so called molecular dynamics (MD) simulation, a method that computes the time evolution of an atomic system by numerically solving Newton's equations of motion. The resulting trajectories provide information about the positions, velocities, and interactions of particles over time, enabling the study of structural and dynamical properties at the atomic scale. To calculate the motion of atoms, the forces acting on each atom at every time step must be known. One approach is to use *ab-initio* methods, which aim to solve the underlying quantum mechanical equations. While these methods are highly accurate, they are also computationally expensive and scale poorly with large systems. To address this challenge, classical force fields were developed. These models use predefined functional forms, which are then parametrized either from experimental data or from *ab initio* calculation results. They are computationally efficient but often lack the accuracy and flexibility needed for more complex or subtle interactions. Since the problem given can be seen as an optimization problem, such that given the atomic positions and types the potential energy / forces of a system should be predicted, there is no inherent need to restrict oneself to a predefined functional form. ML methods offer a promising alternative, bridging the gap between speed and accuracy. The model learns to represent the potential-energy surface (PES) by training on vast amounts of data. This allows MLFFs to achieve *ab initio*-level accuracy at a computational cost similar to that of classical force fields.

2.1 Artificial Neural Networks

Artificial neural networks (ANNs) also just referred to as neural networks (NNs) are biologically inspired algorithms that can be used to model arbitrary functions. Data points that represent the function to be learned are given to the NN, which updates its internal parameters in the training phase. In theory with a suitable NN architecture any function, no matter how complex, can be learned [24]. The basic unit of operation is the artificial neuron that will take a series of inputs x , compute a linear combination

of them, add an offset b and apply a non-linear activation function f_{act} to return the output y (see formula 2.1):

$$y = f_{act} \left(\sum_i w_i x_i + b \right) \quad (2.1)$$

The values w_i are called the weights of the network and together with the offset b , also called bias, they are the parameters that are learned during training. To be able to learn more complex relationships several of these neurons are connected together into a network where information flows between neurons. Depending on how the connections are made several types can be discerned: Residual neural networks (RNN) that are particularly useful for time-series like data, convolutional neural networks (CNNs) usually used in tasks related to image processing or also graph convolutional neural networks (GCNNs) that are very relevant in the field of atomistic simulations.

One of the simplest types of NNs is the fully connected feed-forward neural network (FFNN) where artificial neurons are organized into sequential layers. Each neuron is connected to the neighboring layers, receiving its input from the previous layer and sending its output to the next layer. Connections between neurons in the same layer do not exist, and as such the working of a FFNN can be seen as a stepwise process where information is passed from each layer of artificial neurons to the next, transforming the data in a non-linear way. Data is first passed into an input layer, then through one or more hidden layers until reaching the output layer.

The training of such a network at performing a specific task (e.g. image classification, prediction of material properties) is a cycle consisting of multiple steps. In a first step, also called forward pass, the model is fed with labeled data, transforms its inputs to finally produce a prediction Y_{pred} . In a second step the loss that is to be minimized is computed using a loss function $\mathcal{L}$ that takes Y_{pred} and the known true values/labels Y from the dataset. The loss function is a mean to quantify how well the model performs and guides the entire training process. While the exact functional form of $\mathcal{L}$ depends on the application and other various factors, it is often taken as the squared residual:

$$\mathcal{L}(\{Y_{\text{pred}}\}, \{Y\}) = \sum_{i=1}^n \left(Y_{\text{pred}}^{(i)} - Y^{(i)} \right)^2 \quad (2.2)$$

In a third step, the backward pass, the gradient of the loss with respect to the model parameters is computed. In the case of NNs, this will consist of the weights and biases. The gradient is a measure of how each of the parameters contributes to the loss. Lastly, the parameters are updated based on the loss and gradient of the loss. For this the gradient is passed to an optimizer that takes the current model parameters and the gradient and returns updated parameters that should produce a lower loss. There are a variety of optimizers available, such as gradient descent, stochastic gradient descent [49] or ADAM [27]. This iterative process of forward pass, loss calculation, back-propagation and parameter adjustment is repeated until a satisfactory prediction accuracy is achieved.

2.2 Representation of the PES

First attempts to fit the PES using a non-linear regression model did so by constraining the number of dimensions. In [9] and [35] the adsorption of a small molecule on a surface was carried out by mapping the Cartesian coordinates to a small set of inputs, that represent the symmetry and distance of the adsorbate to the surface. The inputs were then fed into a FFNN with the potential energy E_{pot} as final output. These low-dimensional representations allow for robust and data-efficient training, but are tailored to a specific system and therefore not transferable. For the general case of $3N$ degrees of freedom, several difficulties remain. When using a NN to fit the Cartesian coordinates to E_{pot} , the parameters remain valid only for the specific input size that the model was trained for. Making it usable only for a specific system with a fixed amount of atoms and unchanged elements. Furthermore, the order in which the coordinates are fed into the regression model matter and need to stay fixed. While physically speaking the permutation of 2 atoms of the same element leaves the system indifferent, NNs will have a different output depending on the order. To allow for a high-dimensional PES representation several ML schemes have been proposed like Gaussian processes [4], support vector machines [1], Moment Tensor Potentials [45] or also based on NNs [8, 9]. All of them share the reasonable assumption, that the potential energy E_{pot} of a system can be decomposed into atomic contributions E_i

$$E_{\text{pot}} = \sum_i E_i, \quad (2.3)$$

which in turn only depend on the local chemical environment of the given atoms (all atoms inside a given radius R_c around the central atom). Thus this atom-centered approach [8] changes the original problem of predicting the potential energy of the system, to that of predicting the atomic contributions for each atom in the system. Assuming that the E_i themselves do not depend on indexing of atoms, this solves the problem of permutation invariance. Because the sum of E_i is, by the commutativity property of the addition, independent of the summation order. Also when switching to the atom-centered approach, the potential energy only depends on the relative coordinates, translational invariance is automatically satisfied.

While undoubtedly a better choice than absolute coordinates, the relative Cartesian coordinates still are not suitable as NN input. First, the above mentioned rigidity of the NN input layer does not allow a variable amount of coordinates to be used. Which in turn means that each chemical environment would need a fixed number of neighbors, or in other words a fixed number of atoms within a cutoff radius R_c . It is clear that this condition is unlikely to be satisfied under realistic conditions.

Second, for data efficiency and accuracy, it is desirable that the input fed into the FFNN is invariant to all symmetries of three dimensional space; that is, the euclidean group in three dimensions ($E(3)$). Since the NN output, the atomic energy contribution

E_i , is a scalar quantity and therefore inherently E(3)-invariant, using invariant inputs guarantees that the predicted E_{pot} also respects the those symmetries. Although relative Cartesian coordinates provide translational invariance, as vector quantities they are not invariant under rotations and reflections. For this reason, they cannot be used directly as a NNPs input and need to be transformed into a set of descriptors that is explicitly E(3)-invariant.

Behler-Parrinello symmetry functions

[8] introduced the concept of local descriptors to represent the chemical environment around each atom in a way that is invariant to E(3) symmetries. Their Behler-Parrinello (BP) symmetry functions guarantee a fixed length representation that does not depend on the number of neighbors. This is achieved by using 2 types of symmetry functions:

The first are radial symmetry functions G_i^1 , which encode the pairwise distances between the central atom i and neighbor j . They are defined as a sum of Gaussians shifted by R_s and broadened by η , with the cutoff function $f_c(R_{ij})$ that ensures locality:

$$G_i^1 = \sum_{j \neq i} e^{-\eta(R_{ij}-R_s)^2} f_c(R_{ij}) \quad (2.4)$$

The second are angular symmetry functions G_i^2 , that capture three-body correlations. They are constructed by summing over the cosine values $\theta_{ijk} = \frac{\vec{R}_{ij} \cdot \vec{R}_{ik}}{R_{ij} R_{ik}}$ of the angle spanned by the atoms i, j and k . λ, ζ are again parameters that can be set freely:

$$G_i^2 = 2^{1-\zeta} \sum_{j,k \neq i} (1 + \lambda \cos \theta_{ijk})^\zeta e^{-\eta(R_{ij}^2 + R_{ik}^2 + R_{jk}^2)} f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \quad (2.5)$$

Because of the cutoff function, these functions do not depend on any atom outside of R_c making them local and because they are defined as sums there is no dependency in the order, which provides us with permutation invariance. In practical applications multiple G_i^1 and G_i^2 for different values of η, ζ, R_s and λ will be used, to encode information about the chemical environment around each atom.

Together with the decomposition of E_{pot} into E_i and having rotation, translation and permutation invariant descriptors, using the same NN for each atom paved the way for many other NNPs that follow this design [3, 28, 29, 47].

Spherical Bessel descriptors

Spherical Bessel descriptors constitute a major improvement in the domain of local atomic descriptors. The first generation [28] and then the second generation [29] that additionally is continuous with respect to atomic displacements, provide an efficient, compact and systematic way of describing the chemical environment for use in machine learning potentials (MLPs). First the relative atomic coordinates $\vec{R}_{ij}$ relative to the

central atom i are encoded into a neighbor density function $\rho_i^k(\vec{R})$ (equation 2.6). k is an index for element types in the system, w_j^k a weight factor and δ the Dirac delta function.

$$\rho_i^k(\vec{R}) = \sum_j w_j^k \delta(\vec{R} - \vec{R}_{ij}) \quad (2.6)$$

w_j^k is assumed to be one and not further discussed in [29]. The density functions are projected onto a basis of orthonormal basis functions to yield the complex expansion coefficients c_{inlm} for atom i (equation 2.7).

$$\rho_i(\vec{R}) \approx \sum_{n=0}^{n_{max}} \sum_{l=0}^n \sum_{m=-l}^l c_{inlm} g_{n-l,l}(R) Y_l^m(\theta, \phi) \quad (2.7)$$

g_{nl} is a radial basis function, $Y_l^m(\theta, \phi)$ a spherical harmonic and n_{max} the maximum order of the expansion. To construct g_{nl} , the functions

$$f_{nl}(R) = \sqrt{\frac{1}{R_c^3} \frac{2}{u_{ln}^2 + u_{l,n+1}^2}} \left[\frac{u_{l,n+1}}{j_{l+1}(u_{ln})} j_l \left(R \frac{u_{ln}}{R_c} \right) - \frac{u_{ln}}{j_{l+1}(u_{l,n+1})} j_l \left(R \frac{u_{l,n+1}}{R_c} \right) \right] \quad (2.8)$$

are run through a Gram-Schmidt process for $0 \leq n \leq n_{max}$. $j_l(R)$ is the l -th spherical Bessel function of the first kind and u_{ln} is the $(n+1)$ -th nonzero root of $j_l(R)$. This linear combination of $j_l(R)$ fulfills the requirement $f_{nl}(R_c) = 0$ but also $f_{nl}(R_c)' = 0$, $f_{nl}(R_c)'' = 0$ making the first and second derivative of $g_{nl}(R)$ zero at the cutoff radius. From the radial and angular basis functions in equation 2.7, the expansion coefficients c_{inlm} are calculated from the relative spherical coordinates of neighboring atoms with relation

$$c_{inlm} = \sum_j g_{n-l,l}(R_{ij}) Y_l^{m*}(\theta_{ij}, \phi_{ij}). \quad (2.9)$$

The power spectrum p_{nl} , which represents the final descriptors to be used in a regression model, is then given by

$$p_{inl} = \sum_{m=-l}^l c_{nlm}^* c_{inlm}. \quad (2.10)$$

When substituting equation 2.9 into equation 2.10 the spherical harmonic addition theorem can be applied, reducing the equation to

$$p_{inl} = \frac{2l+1}{4\pi} \sum_j \sum_k g_{n-l,l}(R_{ij}) g_{n-l,l}(R_{ik}) P_l(\cos \gamma_{jik}). \quad (2.11)$$

$P_l(\cos \gamma_{jik})$ is the Legendre polynomial of order l for the angle γ_{jik} spanned by the atoms j , i and k . Equation 2.11 can be evaluated much more efficiently than equation 2.7 on modern computers. Also the former does not need any complex floating point arithmetic even though the expansion coefficients are complex, considerably simplifying the implementation.

2.3 NeuralIL

The NNP used in this work as a foundation for predicting BECs is NEURALIL, originally introduced in [14] and further improved in [38]. This NNP uses a modified form of the spherical Bessel descriptors discussed in section 2.2. The primary modification consists in defining a separate neighbor density function for each element type J using

$$\rho_{iJ}(\vec{R}) = \sum_{\substack{j \in J \\ R_{ij} < R_c \\ j \neq i}} \delta(\vec{R} - \vec{R}_{ij}), \quad (2.12)$$

which applied to equation 2.11 yields the following double sum over pairs of element types J and K :

$$p_{iJKnl} = \frac{2l+1}{4\pi} \sum_{\substack{j \in J \\ j \neq i}} \sum_{\substack{k \in K \\ k \neq i}} g_{n-l,l}(R_{ij}) g_{n-l,l}(R_{ik}) P_l(\cos \gamma_{jik}). \quad (2.13)$$

Compared to the 2nd generation spherical Bessel descriptors (equation 2.11), this formulation produces one channel for each pair of element types. Effectively generating $n_{\text{el}}(n_{\text{el}} + 1)/2$ times more descriptors, with n_{el} being the number of elements in the system. While this quadratic increase of descriptors with element types in the system is undesirable, it has been shown that premixing the descriptors from different element channels decreases prediction performance. Thus implying that premixing the descriptors induces a significant loss of information. The spherical Bessel descriptors only encode information about the chemical environment of an atom.

Because these descriptors only capture structural information around an atom, the element information of the central atom is encoded through an embedding vector of size n_{emb} that is computed from a small FFNN. The embedding vector is concatenated to the spherical Bessel descriptors and form a $E(3)$ invariant representation of the system.

Figure 2.1 shows the complete flow of data. Starting with Cartesian coordinates and atom types on the left to spherical Bessel descriptors and embedding coefficients in an intermediate step, to E_{pot} as a summation over per-atom energies E_i on the right. The underlying FFNN layers use the Swish-1 activation function ($\beta = 1$),

$$s_{\beta}(x) = \frac{x}{1 + e^{-\beta x}} \quad (2.14)$$

which is the result of an automated search for activation functions [41], and has been shown to either outperform or match the rectified linear unit (ReLU) on a variety of datasets. But, more importantly for this work: All Swish activation functions (which are obtained by modifying the β parameter) are twice differentiable everywhere, which is not the case for the heavily used ReLU, whose derivative has a discontinuity at $x = 0$. Combined with the spherical Bessel descriptors this ensures that the entire data-pipeline

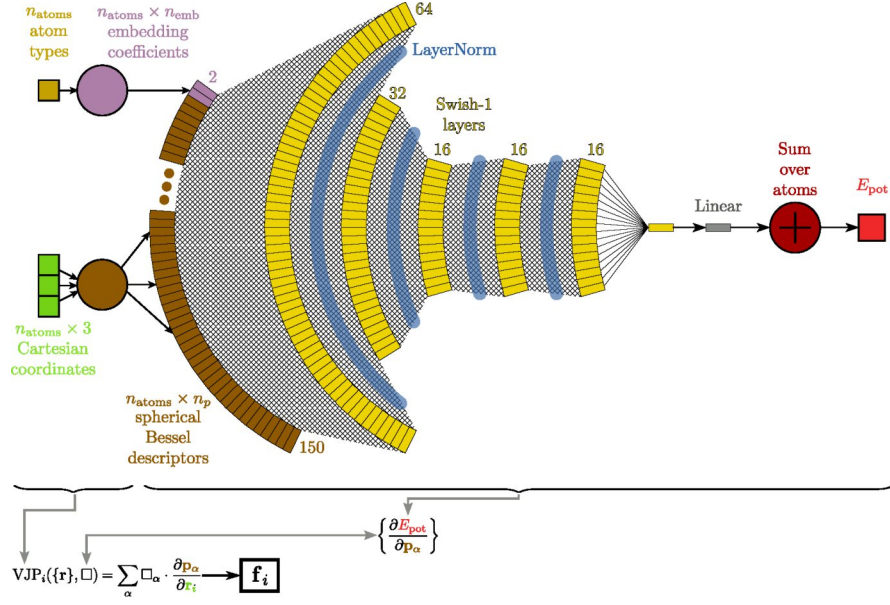


Figure 2.1: Representation of the overall NEURALIL model, including generation of spherical Bessel descriptors, computation of embedding vectors, final sum over atomic contributions to E_{pot} and computation of forces. Taken from [38].

from Cartesian coordinates to E_{pot} is fully differentiable and results in a smooth PES by design.

Forces are obtained as the negative gradient of the network output E_{pot} with respect to the atomic coordinates, $\vec{F}_i = -\nabla_{R_i} E_{\text{pot}}$ computed efficiently using automatic differentiation [7] as implemented in the JAX framework [13]. JAX is a framework for function transformation providing just-in-time compilation (JIT) and automatic differentiation routines. Obtaining forces using differentiation, instead of directly defining a three dimensional vector as NN output, has the advantage of resulting in a force field that is by definition conservative. This ensures that energy is conserved and all forces sum up to zero, eliminating any drift.

Training uses the log-cosh loss with $\langle \cdot \rangle$ defined as the average over structures in a training batch,

$$\mathcal{L} = \lambda_F \left\langle \frac{0.1}{n_{\text{atoms}}} \sum_{i=1}^{n_{\text{atoms}}} \log \left[\cosh \left(\frac{\|\vec{F}_{i,\text{predicted}} - \vec{F}_{i,\text{reference}}\|_2}{0.1 \text{ eV } \text{\AA}^{-1}} \right) \right] \right\rangle + \lambda_E \left\langle 0.02 \log \left[\cosh \left(\frac{E_{\text{pot,predicted}} - E_{\text{pot,reference}}}{0.02 \text{ eV}} \right) \right] \right\rangle \quad (2.15)$$

The log-cosh loss can be seen as a more robust version of the mean average error (MAE). For deviations larger than a characteristic scale parameter ($0.1 \text{ eV } \text{\AA}^{-1}$ for forces and 0.02 eV for E_{pot}) the log-cosh loss behaves similar to MAE, for deviations smaller than the scale parameter it is similar to the mean square error (MSE). The loss function is therefore smooth everywhere and solves the stability problems of MAE. The advantage over directly using MSE is that possible outliers do not get weighted that much, allowing the model to retain consistent performance.

It is worthwhile to note, that not only the direct NN output E_{pot} is included in the loss function, but also the forces which are defined as the negative gradient of E_{pot} . This is an approach known as Sobolev training [15]. From a data-efficiency point of view this is very attractive, since for one structure only one potential energy is given but $3n_{\text{atoms}}$ forces exist. For typical structures, this would result in two or more orders of magnitude more training data when incorporating forces into the loss function. Also, by only training on E_{pot} the model might learn to correctly reproduce the values of the PES, but there is no guarantee that the derivative (forces) are correct. Including the gradient of the target quantity has the effect of making the model more stable and to extrapolate better [2].

Finally, [14] introduced 2 further improvements to NEURALIL:

- The FFNN is replaced with ResNet style layers from [21], which help alleviate optimization difficulties in deep NNs by introducing skip connections.
- Instead of using ADAM [27] as an optimizer, the learned optimizer VELO from [36, 37] is used to accelerate convergence in training.

2.4 Graph convolutional neural networks

So far the ML representation of a high dimensional PES has relied on handcrafted descriptors that encode the local chemical environment of atoms, so that they remain invariant to the symmetries of $E(3)$. However, there is no guarantee that those handcrafted descriptors provide the most efficient encoding of structural information for NNPs.

In recent years GCNN have emerged as a powerful architecture for MLPs, eliminating the need for handcrafted descriptors and allowing the ML model to learn its own representation to encode structural information. GCNNs are the generalization of CNNs to data that is not structured like an euclidean grid. Atomic structures are represented as a graph, of which nodes usually correspond to atoms and the edges to connections to neighboring atoms inside a cutoff radius. Each node/atom has an associated vector of scalar features that is updated through interaction with neighboring atoms. One convolution operation consists of taking the information of neighboring atoms, typically the feature vector of and the scalar distance to the neighboring atoms, and combining them with the current scalar feature vector. By repeating those convolutions information is propagated through the graph, allowing the atoms to build a rich representation of its environments. Since the interatomic distances used in the update of the atoms internal state are rotation invariant, the resulting feature vectors and therefore also the GCNN output are automatically invariant to $E(3)$ symmetries. [6, 42, 43]

Equivariant graph convolutional neural networks

An extension to the GCNN based on rotation invariant features are so called equivariant GCNN. The features are not simply invariant to $E(3)$ symmetries but instead behave in an equivariant way, this means that transforming the system will also transform the features (although not in three dimensional space) in a consistent way [50]. To update the nodes internal state, not only interatomic distances are used, but also the displacement vector between atoms. Equivariance of the features is enforced by constraining the weights of the NN. Unlike the invariant GCNN that discard directional information, having equivariant features enables the ML model to encode more geometric information in the representation. This richer representation of the environment in equivariant GCNN has been shown to be the reason for them achieving state-of-the art performance with good data efficiency [5, 6, 40].

Chapter 3

Born effective charges

To describe electrostatic interactions in materials, several notions of atomic charges have been developed. Static charges, such as e.g. Mulliken charges [39] (based on atomic orbital coefficients) or Bader charges [22] (computed as partitioning of the electron density), provide a conceptually simple way of assigning charges to atoms. However, these static charges are inherently ill-defined quantity that cannot be measured. The problem of assigning static charges to atoms is equivalent the problem of how to partition the electron cloud surrounding the nuclei. Only in the special case where neighboring atoms are clearly separated by a region of zero electron density, one could uniquely assign the electrons to nuclei and therefore also give a unique value for the atomic charge. In most materials atoms are mostly not separated by regions of zero electron density, making it impossible to define a unique atomic charge.

A more physically motivated way of defining charges, is that of the dynamic charge or BEC [17, 18, 48]. The BEC, often abbreviated as Z^* , is a 3×3 tensor defined as the linear response of a system to atomic displacements or external electric fields,

$$Z_{ij}^* = \frac{\Omega}{e} \frac{\partial P_i}{\partial u_j} = \frac{1}{e} \frac{\partial F_i}{\partial \mathcal{E}_j} \quad (3.1)$$

where Ω is the unit cell volume, e the elementary charge, P_i , the component of polarization in direction i , and F_i the component of the force in direction i . Furthermore u_j denotes an atomic displacement and $\mathcal{E}_j$ an electric field, for both quantities the component in direction j . Two equivalent definitions are possible: The change in polarization P_i with the change in atomic position u_j or the change in the force F_i with a change in electric field strength $\mathcal{E}_j$; the indices i and j represent the x, y and z directions. As a practical consequence the electric field induced force $\Delta \vec{F}_{\text{ext}}$ can be computed by simply taking the tensor-vector product,

$$\vec{F}_{\text{total}} = \vec{F} + Z^* \cdot \vec{\mathcal{E}} = \vec{F} + \Delta \vec{F}_{\text{ext}} \quad (3.2)$$

with $\vec{\mathcal{E}}$ an arbitrary electric field and a BEC Z^* . The total force $\vec{F}_{\text{total}}$ is then the sum of the field free forces $\vec{F}$ and $\Delta \vec{F}_{\text{ext}}$. Unlike static charges, which are scalar charges, that

produce only forces parallel to an applied electric field, BECs allow the force response of the system to also have orthogonal components. BECs are therefore a much richer description of electrostatic interactions.

It is not uncommon to see BECs whose values are considerably larger than the corresponding nominal charge. In those instances we speak of *anomalous* BEC and the difference between the BEC and the nominal charge is referred to as *anomalous* contribution. Such large BECs are the result of changes in orbital hybridizations with the accompanying charge redistribution. While for static charges, the charge is seen as localized at the nuclei, BECs directly incorporate the movement of charge that come with hybridization changes as atoms move. Such *anomalous* behavior is often observed in ferroelectric materials [17].

3.1 Computation

BECs are traditionally computed from *ab initio* methods such as density functional theory (DFT). One approach is to use finite differences. Small electric fields in the 3 Cartesian directions are applied and for each the DFT forces are calculated. The partial differential in formula 3.1 is then approximated with a difference quotient. Alternatively density functional perturbation theory (DFPT) can be used to directly obtain BEC without finite differences.

3.2 Prediction using ML

Because *ab initio* methods for calculating BECs are computationally demanding, there is a strong interest in alternative methods that provide accurate BECs at significantly reduced computational cost. Recently, considerable effort has gone into using, similar to MLFFs, machine learning to predict BECs [16, 33, 46]. Three recent strategies for BEC prediction are outlined in the following:

Unified differentiable learning of electric response From the definition of the BEC in equation 3.1 the BEC can be expressed as

$$Z_{ij}^* = -\frac{1}{e} \frac{\partial^2 E_{\text{pot}}}{\partial R_i \partial \mathcal{E}_j} \quad (3.3)$$

which is a mixed derivative of the E_{pot} with respect to the atom position R_i and changes in electric field strength $\mathcal{E}_j$. Since in NNPs forces are often already obtained as a derivative of the potential energy with respect to atomic positions, by extending the derivatives to those in equation 3.3 it should also be possible to calculate BECs [16]. Not only BECs and forces are derivatives of E_{pot} but in the context of electric response also the polarization

$$P_i = -\frac{\partial E_{\text{pot}}}{\mathcal{E}_i} \quad (3.4)$$

and the polarizability, a tensor that indicates how the polarization changes with a change in electric field strength:

$$\alpha_{ij} = \frac{\partial P_i}{\partial \mathcal{E}_j} = -\frac{\partial^2 E_{\text{pot}}}{\partial \mathcal{E}_i \partial \mathcal{E}_j} \quad (3.5)$$

Thus by using derivative relations (see figure 3.1) a NNP can be used to predict various properties in one unified model.

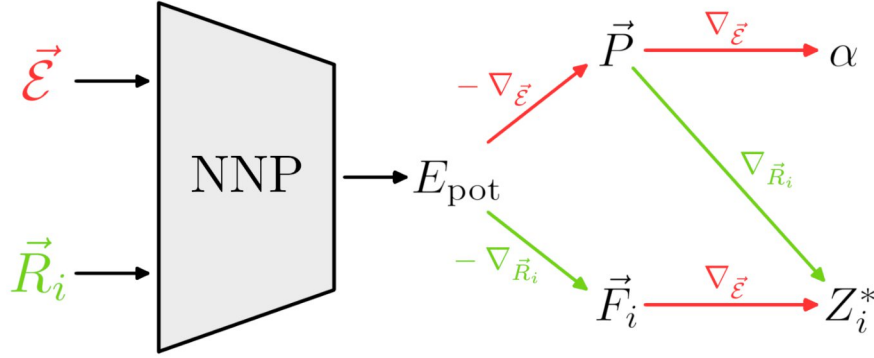


Figure 3.1: Schematic representation of uniform differentiable learning of electric response from [16]. The E_{pot} output of a NNP is differentiated with respect to it's inputs to obtain various properties.

Direct prediction Another approach is the direct prediction of BECs from structural input using NNs, as demonstrated with the graph neural network (GNN) EQUIVAR [33]. Because BEC is a tensor quantity and therefore not invariant to rotations, models that rely on rotation invariant descriptors cannot capture the correct transformation behavior that is essential for tensors. Equivariant GNNs address this issue by explicitly including features that transform with rotation, ensuring that the predicted BECs strictly follow proper transformation rules of tensors in three dimensional space.

Derivation from the polarization A midway approach between direct prediction and prediction with unified differentiable learning of electric response, was proposed in [46]. Here, vector atomic fingerprint descriptors [12] are used as input to a simple FFNN that predicts one component of the polarization vector. Differentiating the NN output derivative with respect to the input positions yields one column of the BEC tensor. This enables the calculation of forces induced by an electric field along one chosen Cartesian direction.

Chapter 4

Methodology

4.1 Training dataset

For the implementation and testing of the framework of BEC prediction a small dataset consisting of 244 water dimers in vacuum was used. The BECs were computed using DFPT, as implemented in VASP package [31, 32] with the projector augmented wave (PAW) [10] formalism. The RPBE exchange-correlation functional [20] was used. After convergence tests an energy cutoff of 1300 eV for the plane wave basis was chosen. Fermi smearing was enabled with $\sigma = 0.1$ eV. As convergence criteria for the SCF-cycle a value of 1×10^{-7} eV was used. The maximum number of electronic self-consistency cycles to 200, all structures converged within that limit. The distribution of BEC values, separated in off-diagonal elements and diagonal elements of the tensor, can be seen in figure 4.1. The water molecule not being an ionic compound, values remain close to zero compared to the formal charges of +1 for hydrogen and -2 for oxygen. Due to its simplicity the water dimer dataset is well suited for the initial development of BEC prediction in NEURALIL. Also the dataset includes several high symmetry structures, which allow to verify that fundamental symmetries are correctly modeled. This can readily be appreciated in figure 4.2.

To also have more realistic test case, models were also trained on a second dataset of Zirconia ZrO_2 , used in [33]. The dataset, which will be referred as ZrO_2 DEFECT, contains a total of 10103 structures spanning four crystal systems. For each system, isotropic strains of -2 %, -1 %, 0 %, 1 %, 2 % were applied, additionally some structures contain a single oxygen vacancy. Representative unit cells are shown in figure A.3 and figure A.4. More details about the composition of ZrO_2 DEFECT are available in table 4.1.

For faster training and testing, a subset, denoted as ZrO_2 NODEFECT, that only contains 394 cubic unstrained $\text{Zr}_{32}\text{O}_{64}$ structures without oxygen defects, was also used. An example structure is shown in figure 4.4 (a).

The distributions of BECs for ZrO_2 DEFECT and ZrO_2 NODEFECT are shown in fig-

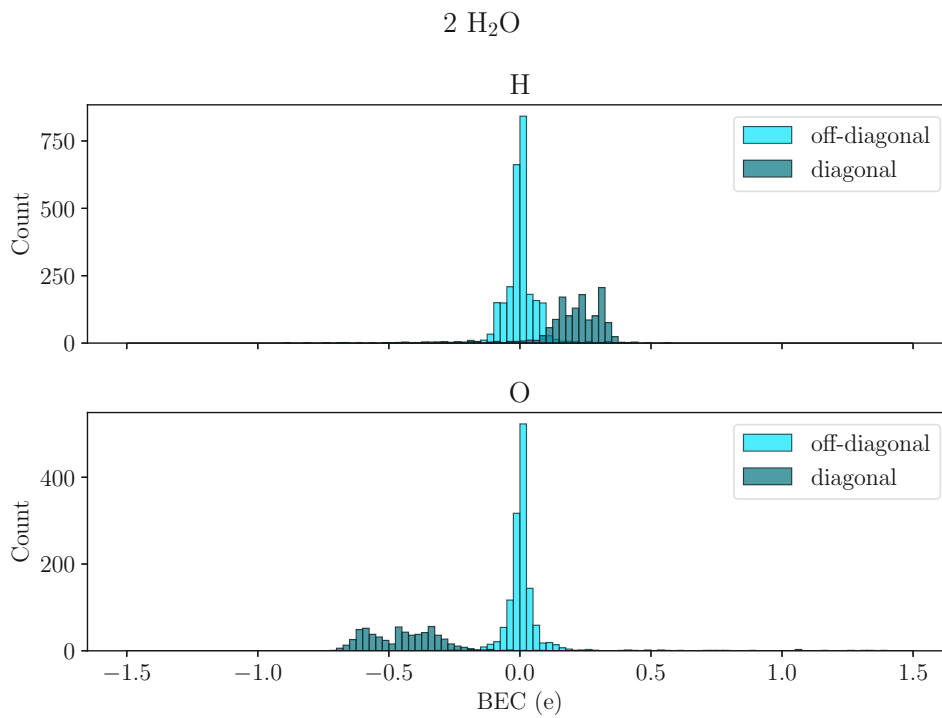


Figure 4.1: Histogram showing the distribution of BECs calculated in VASP for a dataset consisting of 224 water dimers.

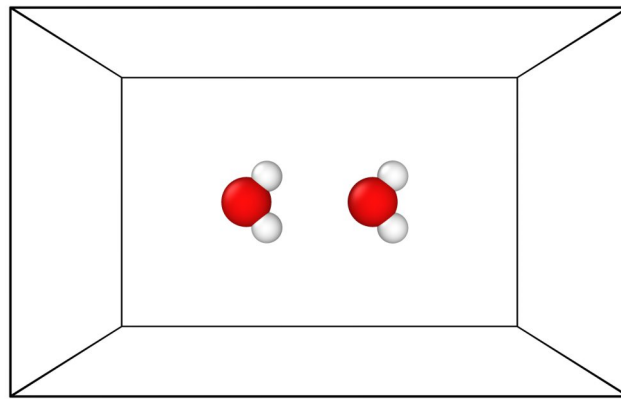


Figure 4.2: One high symmetry structure from the water dimer dataset. All atoms lie inside a plane and a C_2 -axis goes through the oxygen atoms.

Table 4.1: Composition of ZrO_2 DEFECT. Structures sampled from MD runs at 1300 K, 1500 K, 1700 K and 1900 K. All structures come with isotropic variations of the lattice constant (-2% , -1% , 0% , 1% , 2%), some include a single oxygen vacancy.

Crystal system	mp-ID	Structures without defect	Structures with defect
Cubic	mp-1565	1978	1942
Tetragonal	mp-2574	1999	2000
Monoclinic	mp-2858	0	1587
Tetragonal	mp-754403	0	597

ures 4.3 and A.1, respectively. In contrast to the water dimer system, which mainly had covalent bonding, the ionic bonding in ZrO_2 causes the off-diagonal and diagonal BECs to be clearly separated. The off-diagonal components show a broad distribution around zero, indicating that they are not negligible for the dynamics of the system and a treatment of charges as static charges does not fully capture the physics of the system. The arithmetic mean for the diagonal elements of the BEC is approximately $-2.8e$ for oxygen and $5.6e$ for zirconium. Compared to the formal charges of $-2e$ and $4e$ respectively, this highlights the BECs anomalous behavior in zirconia.

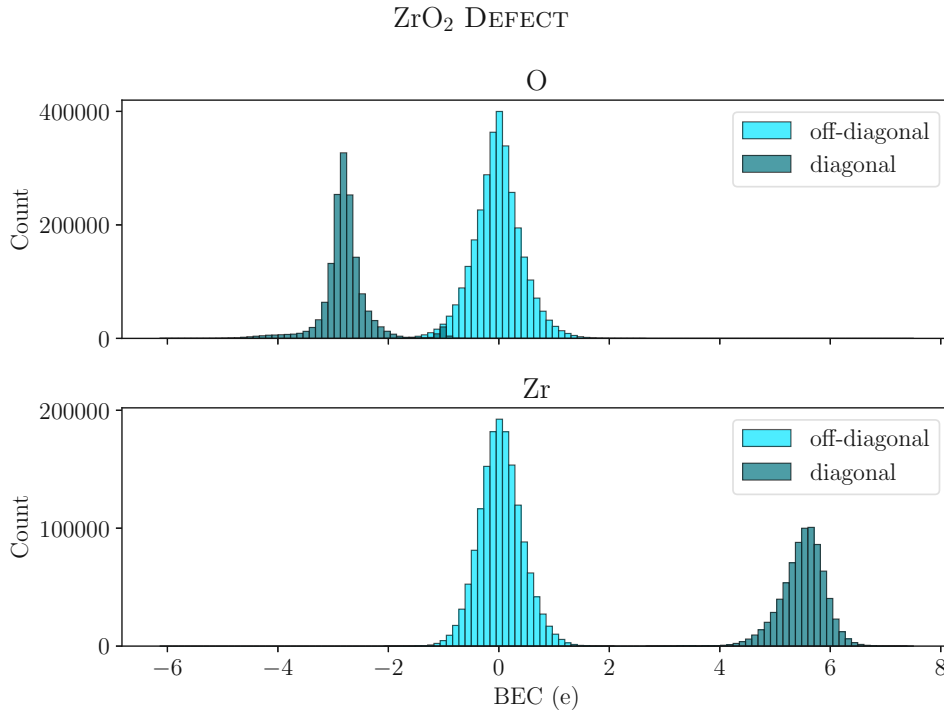


Figure 4.3: Histogram showing the distribution of BECs calculated in VASP for ZrO_2 DEFECT

The third dataset used in this thesis is Li_3PO_4 , taken from [33, 46]. Li_3PO_4 is a widely used solid electrolyte [34]. The dataset consists of 5000 tetragonal pristine $\text{Li}_{12}\text{P}_4\text{O}_{16}$ structures that were sampled from an AIMD run at 2000 K. The distribution of BECs is shown in figure A.2, and one example structure is displayed in figure 4.4 (b).

4.2 Born effective charge prediction using forces

A simplified approach for predicting BECs consists calculating only a single column of the full tensor, as demonstrated in [46]. For many scientifically relevant systems, fixing the electric field in one direction does not consist a serious limitation since this often represents realistic conditions. From formula 3.2 we know that the tensor vector product of a given BEC Z^* and some arbitrary electric field $\vec{\mathcal{E}}$ yields the force $\Delta\vec{F}_{\text{ext}}$ induced by the electric field. By fitting $\Delta\vec{F}_{\text{ext}}$ with neural network force fields (NNFFs)

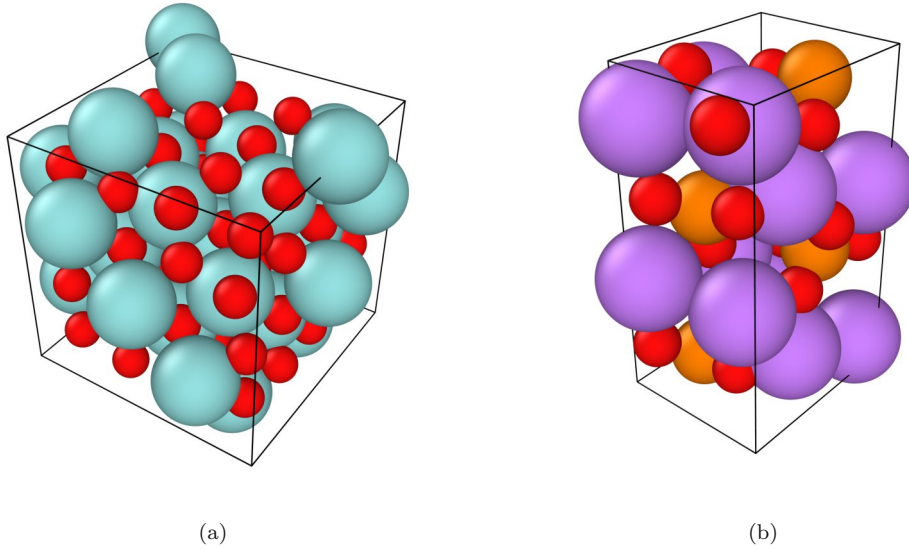


Figure 4.4: (a) Example structure from ZrO₂ NoDEFECT with cubic symmetry. (b) Orthorhombic example structure from the Li₃PO₄ dataset. Phosphorus is represented in orange, lithium in purple.

in a appropriate coordinate system, one is effectively fitting one column of the BEC tensor. However this approach showed to be problematic and a detailed discussion will be given in chapter 5.

Training

To fit $\Delta \vec{F}_{\text{ext}}$ two models were used: NEURALIL [14] and MACE [5]. For NEURALIL n_{max} was set to 6 and ResNet style layers of size [128, 64, 64, 64, 32, 16] in the order input layer to output layer were used. Compared to the standard [64, 32, 16, 16, 16] this might seem exaggerated and prone to overfitting, but it serves as a proof of concept to show whether or not the BEC prediction relying solely of forces is a viable option. The cutoff radius for the local descriptors was set to 3.5 Å enabling each atom in a water molecule to have the remaining 2 atoms in it's receptive field. The dataset was split into 90 % for training and /10 % for validation. Since no E_{pot} was available due to there being no energy that could be assigned to $\Delta \vec{F}_{\text{ext}}$, λ_E in the loss function was set to zero and λ_F to one. The logcosh parameter of the forces was set to 0.2 eV Å⁻¹. Training lasted for 21 epochs, each split into 8 batches and the model with the lowest force root mean square error (RMSE) was saved.

For MACE, which is an equivariant GCNN, the same train test split and number of epochs was used. The hidden irreducible representation parameter was set to "256x0e + 256x1o" meaning that for each node of the graph neural network 256 scalar features and 256 three dimensional vector features are stored. The training lasted for 1000 epochs and the model with lowest force RMSE was saved.

4.3 Born effective charge prediction in unified framework

To make the unified differentiable learning of electric response outlined in [16] work with NEURALIL, 3 important modifications have to be made: First, the model needs to take an additional input, the 3-dimensional electric field vector $\vec{\mathcal{E}}$ to make E_{pot} an electrochemical potential energy [51]. Second, the mixed derivative as defined in equation 3.3 needs to be implemented within the JAX formalism and finally the loss function (equation 2.15) needs to be adapted to include BECs.

Electric field descriptors

Incorporating the electric field $\vec{\mathcal{E}}$ into the model is done indirectly, by defining a function to generate electric field descriptors. In the context of this thesis, electric field descriptors refer to a set of numbers that encodes information about the electric field in a way that is physically meaningful and suitable as input to a NN.

It might not be clear immediately why it is necessary to encode $\vec{\mathcal{E}}$ into a set of descriptors, since it is a fixed size vector and could therefore be fed into the NN directly. But as discussed in section 2.2, the answer lies in the preservation of E(3) symmetries. For example, rotating the entire system including $\vec{\mathcal{E}}$ should not affect the descriptors, whereas rotating only the atomic positions while keeping $\vec{\mathcal{E}}$ fixed should in principle produce different descriptors.

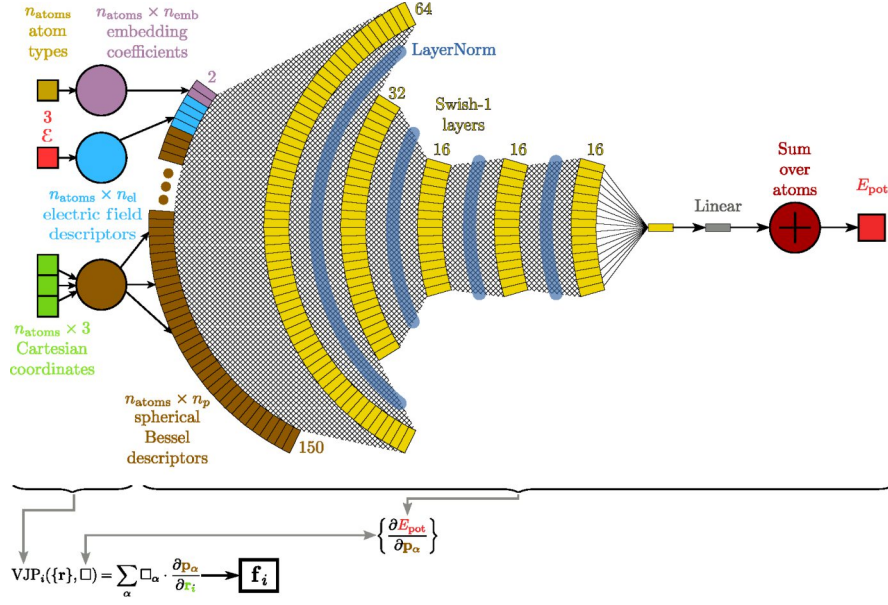
In the modified NEURALIL architecture (figure 4.5), the newly introduced input $\vec{\mathcal{E}}$ is first processed by a descriptor generator function (shown as turquoise circle), which outputs n_{el} electric field descriptors for each of the n_{atom} atoms in the system. These descriptors are then concatenated with the spherical Bessel descriptors and the element embeddings to make up the complete NN input. The specific choice of electric field descriptors is not unique, and the design and analysis thereof constitutes a major part of this thesis.

Identity descriptors As a baseline, the IDENTITY descriptors are defined as the identity operation on $\vec{\mathcal{E}}$, i.e the electric field is directly concatenated to the spherical Bessel descriptors and element embedding:

$$p_i^{\mathcal{E}} = \vec{\mathcal{E}} \quad (4.1)$$

While this choice is straightforward to implement, it fails to satisfy the desired symmetry requirements as discussed in chapter 5.

Radial descriptors A simple set of descriptors that fulfills the basic requirement of rotational invariance is given by the RADIAL descriptors. They are constructed by summing over the dot product of inter atom distances $\vec{R}_{ij}$ with $\vec{\mathcal{E}}$, weighted by radial basis functions $g_{n,l}$ (with $l = 0$), that are also used in the spherical Bessel descriptors.

Figure 4.5: Incorporation of $\vec{\mathcal{E}}$ as an input of NEURALIL. Modified from [38].

The parameter $n_{\text{max}}^{\mathcal{E}}$ determines the maximum order n of the radial basis functions used, and is chosen independently from n_{max} for the spherical Bessel descriptors. $\|\mathcal{E}\|_2^2$ is appended to the $p_{in}^{\mathcal{E}}$ to obtain the electric field descriptors:

$$p_{in}^{\mathcal{E}} = \sum_{j \neq i} \vec{\mathcal{E}} \cdot \vec{R}_{ij} g_{n,0}(R_{ij}), \quad \|\vec{\mathcal{E}}\|_2^2 = \vec{\mathcal{E}} \cdot \vec{\mathcal{E}} \quad (4.2)$$

Radial2 descriptors One drawback of RADIAL is that both the radial basis functions used as weighting as well as the dot product $\vec{\mathcal{E}} \cdot \vec{R}_{ij}$ depend on the magnitude of $\vec{R}_{ij}$. Ideally, distance dependence should be encoded exclusively in the radial basis functions, which were designed for this purpose. To remove the extra distance dependence, a modified version, RADIAL2, that divides the dot product by R_{ij} was developed:

$$p_{in}^{\mathcal{E}} = \sum_{j \neq i} \frac{\vec{\mathcal{E}} \cdot \vec{R}_{ij}}{R_{ij}} g_{n,0}(R_{ij}), \quad \|\vec{\mathcal{E}}\|_2^2 = \vec{\mathcal{E}} \cdot \vec{\mathcal{E}} \quad (4.3)$$

Figure 4.6 illustrates the distance dependence of the two descriptor types RADIAL and RADIAL2 for $n_{\text{max}}^{\mathcal{E}} = 4$. In RADIAL2 (on the left), the division by R_{ij} cancels out the distance dependence of the dot product, leaving only the damped oscillation like behavior from $g_{n,0}$ (the spherical Bessel functions used in the construction of the radial basis functions are indeed solutions that come up in differential equations related to wave propagation). As a result the descriptors become less sensitive when neighboring atoms move further away from the central atom.

In RADIAL (on the right), the extra linear dependence on R_{ij} remains, leading to almost no decrease in magnitude within the cutoff radius R_c . As a consequence, neighboring atoms near R_c influence almost in the same manner than as nearby atoms. For encoding

the structural information the smooth decay present in RADIAL2 is physically more meaningful. Since atoms beyond R_c are ignored it is only natural that atoms close to R_c contribute less to the descriptors, and in a nonlinear way also to E_{pot} .

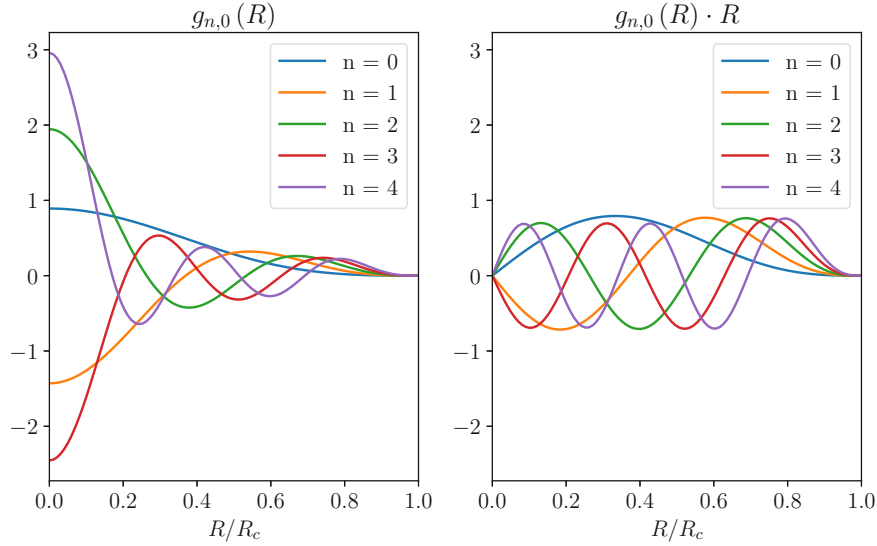


Figure 4.6: Comparison of the effective radial basis used in RADIAL2 on the left and RADIAL on the right.

Angular descriptors A different type of descriptor can be derived by modifying the angular part of the spherical Bessel descriptors. Starting the definition of the spherical Bessel descriptors in equation 4.4, which is equivalent to equation 2.13 with $\cos \gamma_{jik}$ replaced by $\vec{R}_{ij} \cdot \vec{R}_{ik} / (R_{ij} R_{ik})$. $\vec{R}_{ik}$ is replaced by $\vec{\mathcal{E}}$ to yield equation 4.5. This can be interpreted as replacing one of the neighbors in the original double sum with a pseudo atom located at a distance $\vec{\mathcal{E}}$ from the central atom i (figure 4.7), moving from the original double sum over two neighbors (a) to a single sum over one neighbor and a pseudo atom (b).

In practice, since all quantities are evaluated in the limit of $\vec{\mathcal{E}} = 0$, the denominator in equation 4.5 becomes zero, leading to division-by-zero errors. Additionally, the derivative of $g_{n,0}$ at zero is undefined. To circumvent these problems, equation 4.5 is further simplified to equation 4.6. This modification changes the argument of the Legendre polynomial P_l so that it no longer corresponds to the cosine of an angle, breaking the definition of the descriptors as products of expansion coefficients. Nevertheless, despite losing the rigorous derivation, the ANGULAR descriptors are easy to implement, and the NN might still be able to extract more information from ANGULAR than from RADIAL.

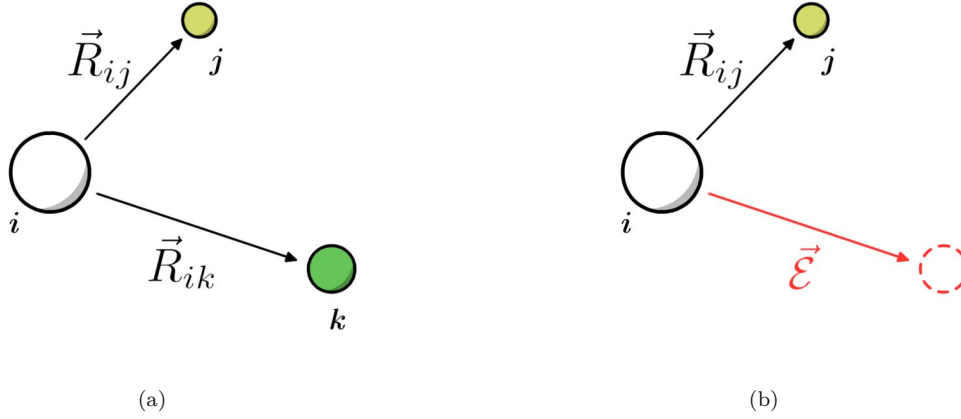


Figure 4.7: Derivation of ANGULAR. One of the neighbors, k , in the double sum (a) is replaced by a pseudo atom at a distance $\vec{\mathcal{E}}$ from i (b).

$$p_{iJKnl} = \frac{2l+1}{4\pi} \sum_{\substack{j \in J \\ j \neq i}} \sum_{\substack{k \in K \\ k \neq i}} g_{n-l,l}(R_{ij}) g_{n-l,l}(R_{ik}) P_l \left(\frac{\vec{R}_{ij} \cdot \vec{R}_{ik}}{R_{ij} R_{ik}} \right) \quad (4.4)$$

$$p_{iJnl}^{\mathcal{E}} = \frac{2l+1}{4\pi} \sum_{\substack{j \in J \\ j \neq i}} g_{n-l,l}(R_{ij}) g_{n-l,l}(\mathcal{E}) P_l \left(\frac{\vec{R}_{ij} \cdot \vec{\mathcal{E}}}{R_{ij} \mathcal{E}} \right) \quad (4.5)$$

$$p_{iJnl}^{\mathcal{E}} = \frac{2l+1}{4\pi} \sum_{\substack{j \in J \\ j \neq i}} g_{n-l,l}(R_{ij}) P_l \left(\frac{\vec{R}_{ij} \cdot \vec{\mathcal{E}}}{R_{ij}} \right) \quad (4.6)$$

OddAngular descriptors The derivative of P_l at zero vanishes for all even values of l . As a direct consequence, all descriptors with even l do not contribute to the prediction of dielectric quantities. To address this, ODDANGULAR, a variant of ANGULAR that only retains odd l , was implemented. This approximately halves the number of descriptors without losing any information.

$$p_{iJnl}^{\mathcal{E}} = \frac{2l+1}{4\pi} \sum_{\substack{j \in J \\ j \neq i}} g_{n-l,l}(R_{ij}) P_l \left(\frac{\vec{R}_{ij} \cdot \vec{\mathcal{E}}}{R_{ij}} \right), \quad l \in 2\mathbb{N} + 1 \quad (4.7)$$

Element descriptors The RADIAL2 descriptors can also be generalized to include element specific information. For each element type J in the system a separate channel is created:

$$p_{iJn}^{\mathcal{E}} = \sum_{\substack{j \in J \\ j \neq i}} \frac{\vec{\mathcal{E}} \cdot \vec{R}_{ij}}{R_{ij}} g_{n,0}(R_{ij}) \quad (4.8)$$

As can be seen from the definition, summing over the different J directly reverts the descriptors back to RADIAL2, with a considerable amount of information lost on the way.

EBessel descriptors Finally, a more direct integration of the electric field into the spherical Bessel descriptors is given in equation 4.9. This form of descriptors, that will be referred to as EBESSEL, includes the electric field as a linear perturbation to the spherical Bessel descriptors. For $\vec{\mathcal{E}} = 0$, they revert back exactly to the modified spherical Bessel descriptors (see equation 2.13) used in NEURALIL, making EBESSEL an augmentation of the original descriptors with electric field sensitivity rather than appending separate descriptors. To allow for varying orders in the encoding the electric field similar to RADIAL, here $n_{\max}^{\mathcal{E}} < n_{\max}$ is defined as the maximum order of n where the linear perturbation term is included in the descriptors. To put it differently, for all n where $n_{\max}^{\mathcal{E}} < n \leq n_{\max}$, equation 2.13 is used instead of 4.9.

$$p_{iJKnl}^{\mathcal{E}} = \frac{2l+1}{4\pi} \sum_{\substack{j \in J \\ j \neq i}} \sum_{\substack{k \in K \\ k \neq i}} g_{n-l,l}(R_{ij}) g_{n-l,l}(R_{ik}) P_l(\gamma_{jik}) \left(1 + \frac{\vec{R}_{ij} \cdot \vec{\mathcal{E}}}{R_{ij}} + \frac{\vec{R}_{ik} \cdot \vec{\mathcal{E}}}{R_{ik}} \right) \quad (4.9)$$

Higher order derivatives in JAX

Since JAX used in NEURALIL is a framework of composable function transformations, is relatively uncomplicated to get higher order derivatives. It only is a matter of combining function transformations. Given a function `calc_potential_energy(positions, types, cell)` that returns the potential energy of a system from atom positions, atom types (elements) and the unit cell size. The gradient with respect to the atomic positions, i.e negative forces, can be calculated with:

```
1 gradient = jax.grad(calc_potential_energy, argnums=0)
```

In NEURALIL, `calc_potential_energy(self, positions, types, cell)` represents the whole NNP, including the generation of spherical Bessel descriptors, the NN layers and sum over individual E_i to give the function output E_{pot} . For the prediction of dielectric properties, the function signature is extended to include the electric field $\vec{\mathcal{E}}$ `calc_potential_energy(self, positions, types, cell, efield)`. At this stage, how $\vec{\mathcal{E}}$ is included as an input, that is, the exact definition of the electric field descriptors is irrelevant, only that $\vec{\mathcal{E}}$ is treated as an additional differentiable input. In the following code snippet the actual function transformations in the modified NeuralIL are shown. `self` indicates an instance of class `NeuralILDielec` which handles learning of dielectric properties:

```
1 # Gradient of Epot with respect to positions
2 self._calc_gradient_pos = jax.grad(self.calc_potential_energy, argnums=0)
3 # Gradient of Epot with respect to efield
4 self._calc_gradient_efield = jax.grad(self.calc_potential_energy, argnums=3)
5 # Function to evaluate Epot and its gradients in one go (gradient with
   positions and efield)
6 self._calc_value_and_gradients = jax.value_and_grad(
7 self.calc_potential_energy, argnums=[0,3]
8 )
9 # Second derivative of Epot with efield
```



```

10 self._calc_hessian_efield = jax.hessian(self.calc_potential_energy, argnums
    =3)
11 # Mixed derivative of Epot with positions and efield
12 self._calc_jacobian_gradientpos_efield = jax.jacfwd(
13 self._calc_gradient_pos, argnums=3
14 )

```

Note that `jax.grad()` performs reverse-mode automatic differentiation, which is more efficient when the target function has more inputs than outputs. This is ideal for the function computing E_{pot} since it only has one scalar output. The computation of BECs requires a mixed derivative with respect to atomic positions and the electric field (equation 3.3). This is achieved by first applying `jax.grad()` on `calc_potential_energy` for the second argument, effectively generating `self._calc_gradient_pos` a function that produces gradients of the E_{pot} with respect to the positions. `self._calc_gradient_pos` is then differentiated with respect to $\vec{\mathcal{E}}$ using `jax.jacfwd` which performs forward-mode automatic differentiation. Forward-mode automatic differentiation is more efficient for functions where the dimension of the inputs is similar or smaller than that of the outputs, which is true for `self._calc_gradient_pos`. Thus, combining reverse-mode automatic differentiation with forward-mode differentiation of `calc_potential_energy` allows to obtain BECs efficiently.

Loss function

The loss function from equation 2.15 is extended with an additional term, $\Delta\mathcal{L}$ (equation 4.11), which accounts for quantities related to the electric field. Here, $\|\cdot\|_F$ denotes the Frobenius norm. Each of the batch averages ($\langle \cdot \rangle$) is weighted with the corresponding weight factor (λ_Z , λ_P and λ_α for BEC, polarization and polarizability respectively). To account for the multivaluedness of the polarization in periodic systems, as described in the modern theory of polarization [48], the polarization loss is defined with the minimal image convention. An integer vector $k \in \mathbb{Z}^3$ is chosen such that

$$\Delta\vec{P}_{\text{MIC}} = \min \left| \vec{P}_{\text{predicted}} - \vec{P}_{\text{reference}} + \text{diag}(k)\Delta\vec{P}_q \right| \quad (4.10)$$

is minimized, where the polarization quantum $\Delta\vec{P}_q$ is defined as the product of the unit cell size $\vec{a}$ with the elementary charge e . This can alternatively be regarded as a folding of $\vec{P}_{\text{predicted}} - \vec{P}_{\text{reference}}$ to the range $[-0.5\Delta\vec{P}_q, +0.5\Delta\vec{P}_q]$.

The logcosh parameters for each of the quantities in the loss function are chosen to approximately match the RMSE of that quantity in the final trained model. As already noted in [38], the exact value of these parameters has a negligible effect on performance.

This could be confirmed again in hyperparameter tests.

$$\begin{aligned} \Delta\mathcal{L} = & \lambda_Z \left\langle \frac{0.1}{n_{\text{atoms}}} \sum_{k=1}^{n_{\text{atoms}}} \log \left[\cosh \left(\frac{\|Z_{k,\text{predicted}}^* - Z_{k,\text{reference}}^*\|_F}{0.1 e} \right) \right] \right\rangle + \\ & \lambda_P \left\langle 0.05 \log \left[\cosh \left(\frac{\|\Delta\vec{P}_{\text{MIC}}\|_2}{0.05 e \text{ \AA}} \right) \right] \right\rangle + \\ & \lambda_\alpha \left\langle 0.03 \log \left[\cosh \left(\frac{\|\alpha_{\text{predicted}} - \alpha_{\text{reference}}\|_F}{0.03 e \text{ \AA}^2 \text{ V}^{-1}} \right) \right] \right\rangle \end{aligned} \quad (4.11)$$

Training - water dimer

For water dimer dataset the loss weights λ_Z , λ_α , λ_F , λ_P , λ_E in the modified loss function $\mathcal{L} + \Delta\mathcal{L}$ were set to 10, 10, 2, 2, 1 respectively, assigning higher weights for quantities representing higher-order derivatives in accordance to [16]. The parameters of the log cosh loss that determines the change from quadratic to linear behavior were set to 0.05, 0.03, 0.15, 0.05, 0.02. The cutoff radius was 3.5 Å, with $n_{\text{max}} = 4$ and the default NN layers of [64, 32, 16, 16, 16]. Again a dataset split of 90 % training dataset and 10 % validation dataset was used. The training was run for 200 epochs, with the data split into 8 batches.

Training - zirconia

Since the dataset ZrO₂ DEFECT dataset does not include polarizations and polarizabilities, the corresponding weights in the loss function were set to zero. The parameters of the log cosh loss were set to 0.1 for BEC, 0.02 for energies and to 0.1 for forces. An extensive hyperparameter testing was conducted using the smaller and therefore faster to train on ZrO₂ NODEFECT dataset. The tests showed that the previously used $r_{\text{cut}} = 3.5$ Å was not sufficient, which was to be expected from the bigger interatomic distances. A cutoff radius of 4.3 Å proved to be a good compromise between performance and accuracy. Increasing n_{max} to 8 gave significant performance gains, the number of spherical Bessel descriptors with that parameter amount to 135, which is comparable to the 150 in [38]. The number of training epochs was increased to 600. While the train/validation split was left unchanged, with the increased dataset size the number of batches was further increased to 16. Also a diversity of different layer widths and depths were tested for the NN but the default choice remained the best. Deeper architectures showed unstable convergence and were prone to overfitting while wider architectures showed no improvement at all at the cost of a larger model. With higher order derivatives and more data to fit in general it would have been thinkable that a deeper network or one with more parameters performs better which interestingly was not the case.

Chapter 5

Results

5.1 Born effective charge prediction using forces

Fitting forces induced by an electric field, rather than fitting BECs directly, has the advantage of requiring no modifications to NNFFs. In the current approach, forces induced by 3 electric fields, $\vec{\mathcal{E}}_x$, $\vec{\mathcal{E}}_y$ and $\vec{\mathcal{E}}_z$, each with an electric field strength of $2.57 \text{ V } \text{\AA}^{-1}$ and pointing in the x , y and z direction respectively are computed from the BECs and fitted using an NNFF. The resulting fit of the forces induced by $\vec{\mathcal{E}}_z$ can be seen in figure 5.1. Based on this parity plot alone one might come to the conclusion that predicting BECs indirectly using forces is a viable strategy. However, that those forces are fitted correctly is more of a coincidence. This becomes clear when examining figure 5.2 which shows the parity plots for forces induced by $\vec{\mathcal{E}}_x$ and $\vec{\mathcal{E}}_y$ respectively, both making it evident that there is a fundamental problem with this approach.

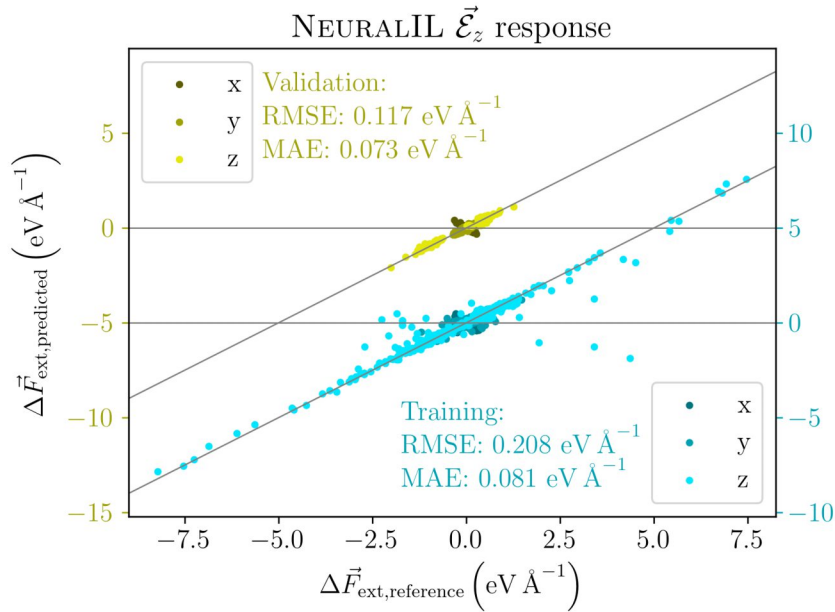


Figure 5.1: Fitting of forces $\Delta \vec{F}_{\text{ext}}$ induced by the electric field $\vec{\mathcal{E}}_z$ using NEURALIL.

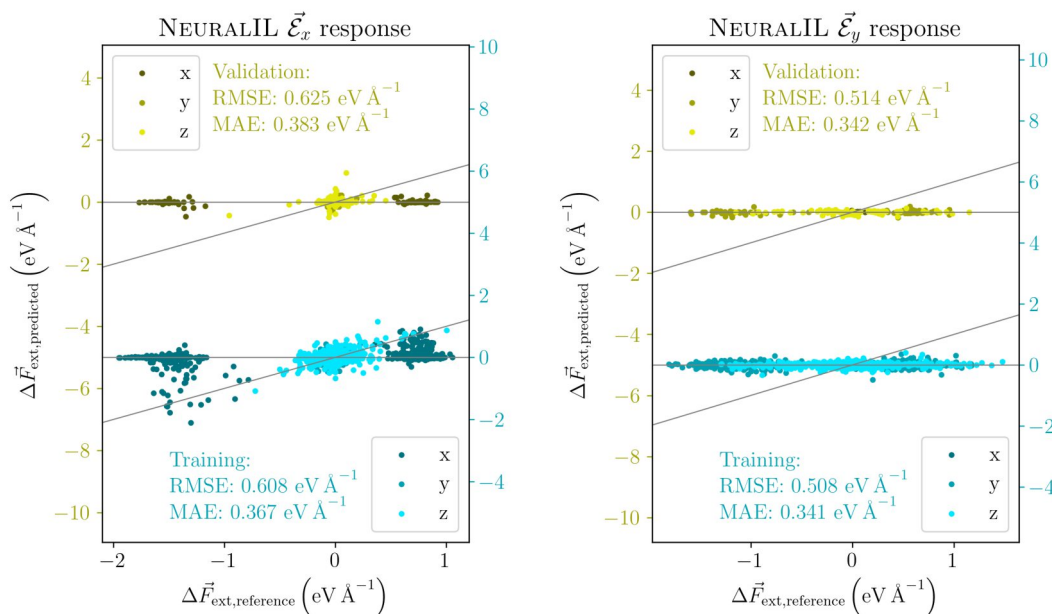


Figure 5.2: Fitting of forces $\Delta \vec{F}_{\text{ext}}$ induced by electric fields $\vec{\mathcal{E}}_x$ and $\vec{\mathcal{E}}_y$ using NEURALIL.

The same fitting procedure was also applied to the NNFF MACE [5], showing very similar results (see figures B.1 and B.2). To get a better understanding of the underlying problem, one highly symmetric structure (figure 4.2) and the corresponding forces were analyzed.

The forces $\Delta \vec{F}_{\text{ext}}$ induced by $\vec{\mathcal{E}}_z$, $\vec{\mathcal{E}}_y$ and $\vec{\mathcal{E}}_x$ can be seen schematically for one of the water molecules in the high symmetry structure (see figure 4.2) in figure 5.3. First taking a closer look to the case (b), where the electric field $\vec{\mathcal{E}}_y$ is applied: The force on the oxygen atom points exactly in the opposite direction to the electric field vector while for the 2 hydrogen atoms the forces point approximately in the same direction as the electric field. Although there is nothing inherently wrong with those forces, it is important to note that the NNFF has only Cartesian coordinates and atom types as inputs and therefore has no knowledge of the external electric field. From the perspective of the NNFF, the molecule is still a perfectly symmetric H_2O molecule, even though in reality the applied electric field $\vec{\mathcal{E}}_y$ breaks the horizontal mirror symmetry of the xy -plane.

By design NEURALIL and many other NNFFs respect such symmetries. This means E_{pot} is invariant under those symmetry operations and the forces transform as normal vectors in three dimensional space. Because the oxygen lies in the xy -mirror plane, its y -force component has to be equivalent to its mirror, that is, the negative y -force component, effectively rendering it zero. Similarly the forces acting on the two hydrogen atoms also need to be the mirror images of each other. Performing a mirror operation followed by a sign flip yields the corresponding other hydrogen atoms force. Clearly the forces induced by $\vec{\mathcal{E}}_y$ do the opposite of what symmetry demands. All structures in the dataset, while not perfectly symmetric, have a similar orientation to the example provided before. It

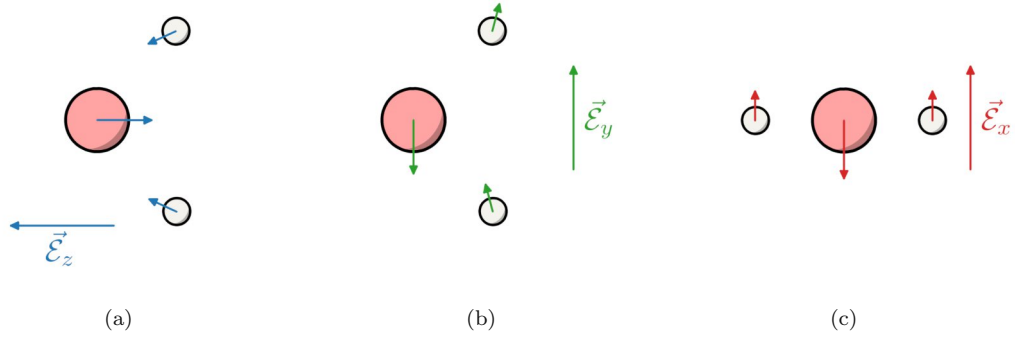


Figure 5.3: Induced forces $\Delta\vec{F}_{\text{ext}}$ shown schematically for (a) $\vec{E}_z$, (b) $\vec{E}_y$ and (c) $\vec{E}_x$.

is not surprising to see results like in figure 5.2 where almost all predictions are close to zero. The model cannot reproduce forces that violate the symmetry it enforces.

A similar situation arises in case (c) $\vec{E}_x$: All atoms lie in the yz mirror plane and the electric field is perpendicular to that. If the electric field is absent, from symmetry, the x -components of all forces must be zero. Applying an electric field breaks this symmetry, but here again, this is not something the NNFF can see, which results in close to zero predictions for all forces as can be seen in figure 5.2.

In contrast, forces induced by $\vec{E}_z$ are fitted reasonably well. This can be explained again with symmetry. The electric field vector is parallel to both mirror planes, therefore no symmetries are broken and the forces remain consistent with the models implicit enforced symmetry.

5.2 Born effective charge prediction in unified framework

In the previous section it was shown that predicting BECs indirectly via forces is not a viable option. The problem arises from the NNFF that was used respecting the fundamental symmetries of $E(3)$. In principle a NNFF that does not respect those symmetries such as ForceNet [25] could be used. However, this would require considerably more training data and the the model would have to explicitly learn how rotation affects forces.

A more efficient approach is the unified differentiable learning of electric response [16], that was implemented in NEURALIL as part of this thesis. As explained in chapter 4, three main modifications were necessary:

- Adding the electric field vector as a input to NEURALIL.
- Implementing the corresponding derivatives of E_{pot} to obtain the polarization, polarizabilities and BECs.
- Changing the loss function to also include polarization, polarizabilities and BECs.

To include $\vec{\mathcal{E}}$ as an input first the IDENTITY descriptors introduced in chapter 4 were first used. IDENTITY only performs the identity operation on the electric field before concatenating it with the other descriptors. In other words, the electric field vector is appended to the remaining descriptors without any modification. The performance of the trained model is shown in figure 5.4 for E_{pot} and forces. For the small dataset of water dimers considered here, the predictions are highly accurate, as expected. Since these quantities are not affected by an electric field of zero, the results essentially reflect the performance of the unmodified NEURALIL. For the prediction of BECs the modifi-

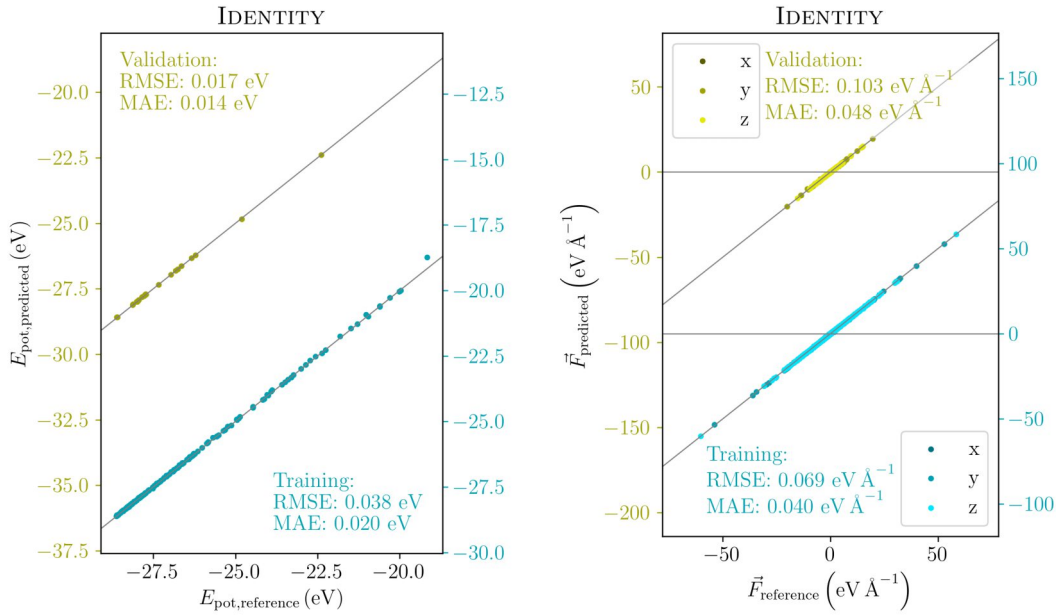


Figure 5.4: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the water dimer dataset using the unified differentiable learning of electric response framework with the IDENTITY descriptors.

ations added in this thesis become relevant. The corresponding parity plot for BECs is shown in figure 5.5. The diagonal elements of Z^* denote the response of the system in the same direction as the perturbation whereas off-diagonal elements represent the system's response in orthogonal directions to the perturbation. Therefore in most systems the diagonal elements tend to be much larger than the off-diagonal elements. To reflect this in the parity plots of BECs, diagonal components and off-diagonal components of the tensor are discerned by assigning different colors.

When predicting BEC with IDENTITY descriptors, we can see that some diagonal components get predicted close to zero by the model. Upon closer examination it becomes apparent that the Z_{xx}^* , Z_{yy}^* are the problematic points. This is also the same direction that also couldn't be predicted in forces-only approach discussed in section 5.1. Similarly the polarization predicted by the model with IDENTITY descriptors fails for the x and y -components, but accurately captures the z -components. In contrast, the predicted polarizabilities show good agreement with the DFPT ground truth (figure B.3). Considering that the water dimer dataset is very small and composed only of highly

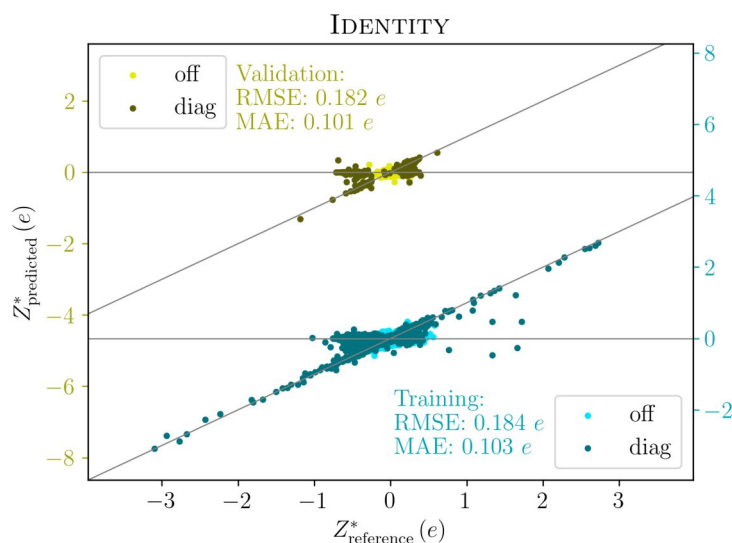


Figure 5.5: Prediction of BECs in the water dimer dataset using the unified differentiable learning of electric response framework with the IDENTITY descriptors.

similar structures, it is unlikely that the model itself is not powerful enough to describe the system. Instead these shortcomings hint to a more fundamental limitation of the IDENTITY descriptors.

As discussed in chapter 2, descriptors should provide physically meaningful representation of an atom’s environment. For NNPs we required the descriptors to be invariant to rotation, translation and permutation of atoms. How exactly to extend these requirements to descriptors that encode electric field information is best explained with an example.

The upper half of figure 5.6 shows a water molecule with some applied electric field $\vec{\mathcal{E}}$ together with the data flow to the final NNP output E_{pot} . The embedding vector (purple) only encodes the element type of the central atom. The Power spectrum generator calculates the 2nd generation spherical Bessel descriptors (brown), which depend solely on atomic positions and element types of the neighbor atoms. Finally, the electric field descriptors (turquoise), whose functional form is yet to be defined, are intended to encode the electric field information. These three types of descriptors concatenated are fed into the NN to yield atomic energies, which are the summed to obtain E_{pot} .

The expected behavior under a uniform rotation of both the atomic positions $\vec{R}_i$ and $\vec{\mathcal{E}}$ are shown in the bottom half of figure 5.6. Physically speaking, such a rotation leaves the system itself unchanged, only the observer’s viewpoint changes. Therefore all descriptors should stay unchanged and as a direct consequence E_{pot} invariant. This requirement immediately invalidates the IDENTITY descriptors. Because they depend directly on $\vec{\mathcal{E}}$, any change in magnitude or direction is going to alter the descriptor values. In the example shown, rotating the whole system changes the direction of $\vec{\mathcal{E}}$. As a result the descriptors change along with the predicted E_{pot} even though it must

remain unchanged. While the spherical Bessel descriptors and embedding vector preserve rotation invariance, IDENTITY descriptors violate it, making them fundamentally unsuitable as electric field descriptors.

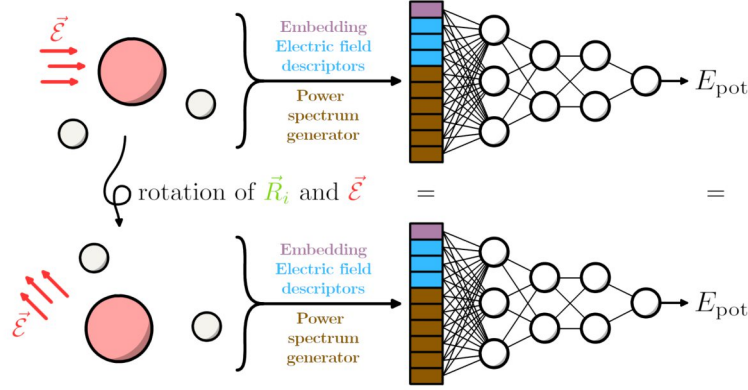


Figure 5.6: Depiction of how ideally the descriptors used in the NNP should behave under rotation of both positions $\vec{R}_i$ and electric field vector $\vec{E}$.

One might consider using a descriptor that depends only on the magnitude $\|\vec{E}\|_2$ of the electric field. This is possible in principle and avoids the rotational invariance problems described above. However, BECs are defined in the limit of zero electric field, thus the electric fields used during model training and validation are three element null vectors. The quantities of interest BEC, polarizations and polarizabilities are obtained as derivatives of E_{pot} with respect to $\vec{E}$ at $\vec{E} = 0$. As a result we need differentiability of E_{pot} with respect to $\vec{E}$ which is not given for $\|\vec{E}\|_2$. Using instead $\|\vec{E}\|_2^2$ would restore differentiability at $\vec{E} = 0$, but render all derivatives 0, leaving the model unable to predict anything else that $\vec{P} = 0$, $Z_i^* = 0$ and $\alpha = 0$. This is already an indication that the electric field descriptors cannot depend solely on $\vec{E}$; they must take into account atomic positions as well.

Returning to the water molecule example, we can consider the case where only the

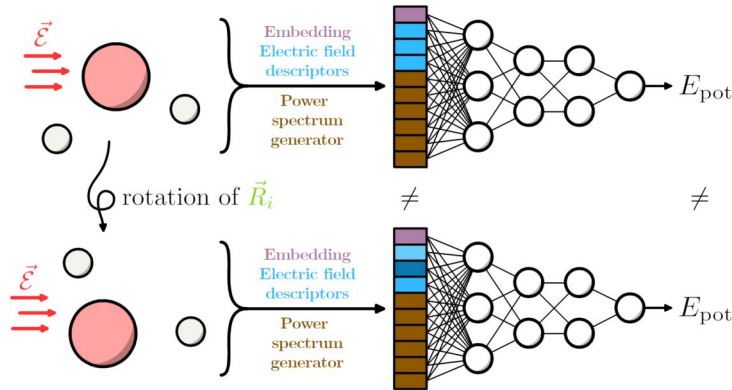


Figure 5.7: Depiction of how ideally the descriptors used in the NNP should behave under rotation of positions $\vec{R}_i$.

positions are rotated while the electric field vector remains fixed in space (see figure 5.7). From the molecules perspective, $\vec{\mathcal{E}}$ now points in a different direction. This very likely leads to an altered electron density. Since the electron density uniquely defines the ground-state wavefunction [23] and thus energy, this transformation does not leave the physical system unchanged. The embedding vector and spherical Bessel descriptors do not reflect this change, leaving only the electric field descriptors to distinguish the transformed system from the original system. However IDENTITY descriptors remain unchanged if $\vec{\mathcal{E}}$ remains the same, again failing to capture the physics. The same limitation also applies to descriptors based on $\|\vec{\mathcal{E}}\|_2$ or powers thereof, which would also not be able to distinguish between these physically distinct configurations.

To make the electric field descriptors invariant under rotation, similar to the example shown above, an approach analogous to the spherical Bessel descriptors and other local descriptors can be adopted. Such descriptors only depend on interatomic distances and the angles spanned by 2 neighboring atoms and the central atom (angle located at central atom). Because these quantities are independent of the absolute orientation of the system, they naturally are rotation invariant. The proposed RADIAL descriptors (see equation 4.2) implement this by taking dot products of $\vec{\mathcal{E}}$ and interatomic distances $\vec{R}_{ij}$. It might not be obvious on a first glance why $\vec{\mathcal{E}} \cdot \vec{R}_{ij}$ is an appropriate choice for enforcing rotation invariance, but using the cosine relation

$$\vec{\mathcal{E}} \cdot \vec{R}_{ij} = \|\vec{\mathcal{E}}\|_2 \|\vec{R}_{ij}\|_2 \cos \theta \quad (5.1)$$

with θ as the angle between the two vectors, it follows that the dot product only depends on the magnitude of and angle between the vectors. A new parameter $n_{\max}^{\mathcal{E}}$ is introduced which determines the number of radial basis functions to use for the electric field descriptors, chosen independently of $n_{\max}$ for the spherical Bessel descriptors.

Using $n_{\max}^{\mathcal{E}} = 5$ and training on the water dimer dataset, the prediction of quantities related to the electric field, namely BECs, polarizations and polarizabilities show good accuracy. For BEC prediction (figure 5.8), predicted values are in close agreement with the reference values for both diagonal and off-diagonal tensor components. While no major improvements are observed for polarizabilities, predicted polarizations are now considerably more accurate and do not show artifacts such as x and y -components being predicted close to zero (figure B.5). As expected, energy and force predictions (figure B.4) remain comparable to those with IDENTITY descriptors.

ZrO₂ NoDefect With descriptors at hand that respect symmetry requirements, it makes sense address a more complex dataset. Using the RADIAL descriptors, the BECs of ZrO₂ NoDEFECT can be predicted accurately. Unlike the water dimer, zirconia is an ionic compound and therefore exhibits a clear separation of diagonal from off-diagonal tensor components. Setting $n_{\max}^{\mathcal{E}}$ to 11 shows good performance. With that $n_{\max}^{\mathcal{E}} + 2 = 13$ RADIAL descriptors are generated. Compared to over 100 spherical Bessel descriptors

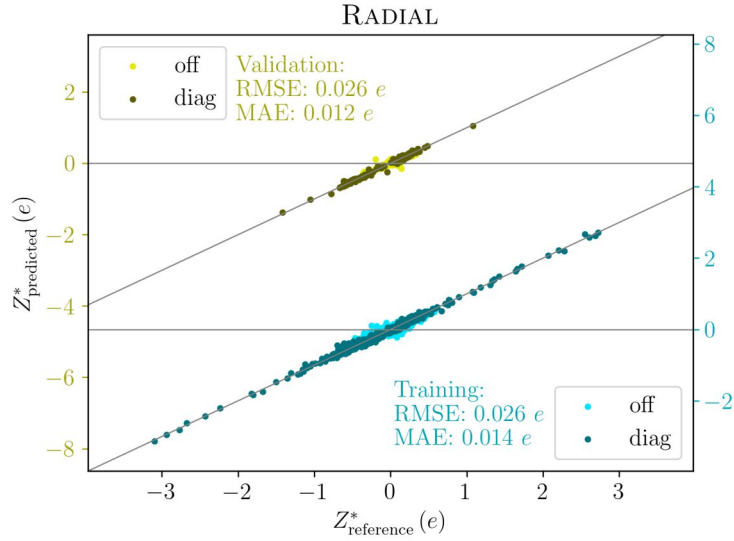


Figure 5.8: Prediction of BECs in the water dimer dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

for $n_{\max} = 8$, the impact on inference time is clearly negligible. Further increasing $n_{\max}^{\mathcal{E}}$ showed no increase in prediction accuracy. With this parameters, we obtain predictions close to the reference values for BECs (figure 5.9) and energy/forces (figure B.6).

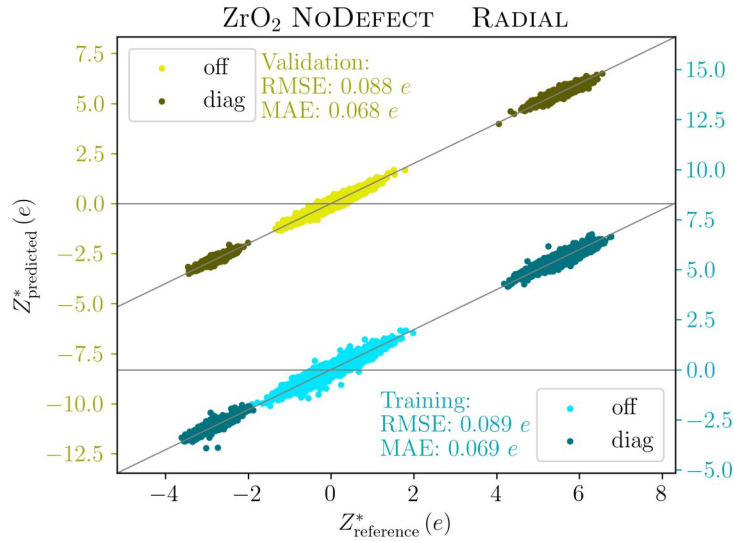


Figure 5.9: Prediction of BECs in the ZrO₂ NoDEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

Also alternative descriptors were tested using the same parameters. The RADIAL2 descriptors, where the distance dependence is isolated in the radial basis functions and that were expected to show superior performance, showed no significant difference to RADIAL. Also ELEMENT descriptors were used with $n_{\max}^{\mathcal{E}} = 11$. For a system with 2 types of elements such as zirconia, the splitting of descriptors into element channels

results in twice as much descriptors as with RADIAL. As summarized in table 5.1, the ELEMENT descriptors had an impact on performance: while force RMSE remained unchanged, the BEC RMSE decreased, indicating that element-specific channels help to encode more information about the electric field.

Table 5.1: Final validation set performance of NEURALIL models on the ZrO_2 NODEFECT dataset. From six trained models the mean performance and the one of the model showing lowest Z^* RMSE is shown.

	Arithmetic mean of RMSE		RMSE of model with lowest Z^* RMSE	
	Z^* (e)	$\vec{F}$ (eV Å ⁻¹)	Z^* (e)	$\vec{F}$ (eV Å ⁻¹)
RADIAL	0.097	0.080	0.092	0.078
RADIAL2	0.097	0.082	0.093	0.078
ELEMENT	0.091	0.078	0.088	0.078

ZrO₂ Defect Table 5.2 and figure 5.10 show the final validation set performance of several NEURALIL models trained on the ZrO_2 DEFECT dataset. Training runs that diverged were started again so that in total four converged models for each descriptor type or other modification are available. RADIAL descriptors served as baseline to compare with the descriptors designed in this thesis. Although in principle the choice of electric field descriptors should not influence force predictions, in practice we observe a correlation. Models that have low BEC RMSE also exhibit higher force RMSE.

Table 5.2: Final validation set performance of NEURALIL models on the ZrO_2 DEFECT dataset. From four trained models the mean performance and the one of the model showing lowest Z^* RMSE is shown. Models marked with * are trained with a log cosh parameter of 10 e for the BEC and $\lambda_{\text{BEC}} = 20$.

	Arithmetic mean of RMSE		RMSE of model with lowest Z^* RMSE	
	Z^* (e)	$\vec{F}$ (eV Å ⁻¹)	Z^* (e)	$\vec{F}$ g(eV Å ⁻¹)
RADIAL	0.183	0.118	0.179	0.115
ANGULAR	0.197	0.114	0.160	0.122
ODDANGULAR	0.181	0.123	0.170	0.119
ELEMENT	0.184	0.123	0.156	0.112
ELEMENT*	0.183	0.102	0.150	0.101
EBESSEL	0.143	0.154	0.132	0.156
EBESSEL*	0.157	0.116	0.150	0.120

The RADIAL descriptors (with $n_{\text{max}}^{\mathcal{E}} = 11$) achieve BECs predictions with a RMSEs deviation of around 4 % relative to a reference value of 5 e. In itself, this accuracy could already be sufficient for MD simulations under electric field, since forces induced by the electric field under realistic conditions are at least one order of magnitude smaller than the field-free forces. At this level of accuracy for BECs, deviations from AIMD will mostly be due to errors in the field-free force predictions themselves. However, upon closer inspection we can see problematic artifacts: BEC predictions for some oxygen atoms have unphysical values up to 10 e instead of the expected $-2.5 e$ (figure 5.11). Although these outliers, that is, those datapoints that lie inside the rectangle from

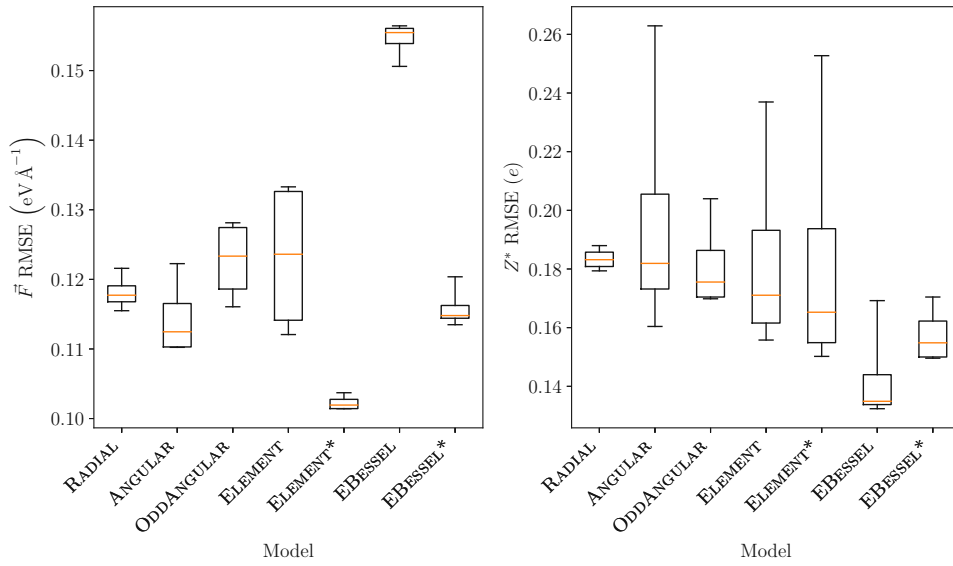


Figure 5.10: Final validation set performance of NEURALIL models on the ZrO_2 DEFECT dataset. Visual representation of data in table 5.2. Models marked with * are trained with a log cosh parameter of $10 e$ for the BEC and $\lambda_{\text{BEC}} = 20$.

$(-5 e, 0 e)$ to $(-1.16 e, 10 e)$, amount to ~ 1000 , which is a very small fraction of the ~ 7 million data points, they can alter the system dynamics in critical moments of a MD simulation. Interestingly, the force and energy predictions in the same model do not show any of these artifacts (see figure 5.12), suggesting that RADIAL descriptors fail to encode the electric field information adequately. This limitation motivated the development of descriptors beyond IDENTITY and RADIAL.

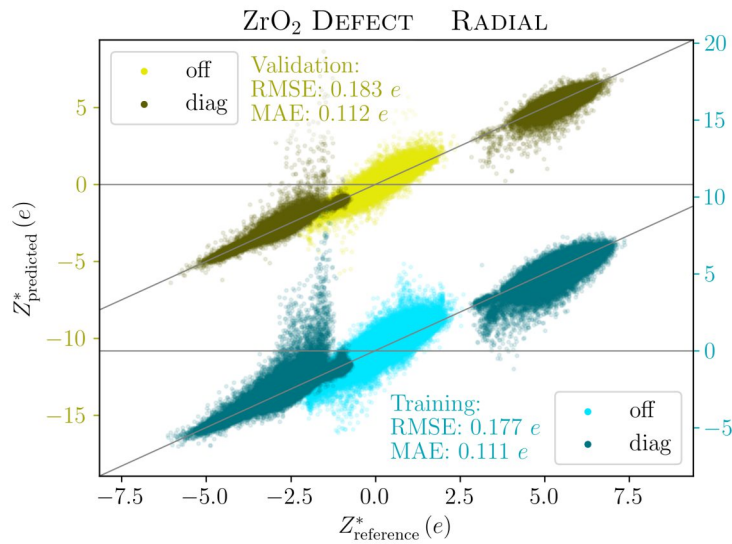


Figure 5.11: Prediction of BECs in the ZrO_2 DEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

The ANGULAR descriptors (with $n_{\text{max}}^{\mathcal{E}} = 8$) show inconsistent results. While their best performing model outperforms RADIAL, the variability is very big. Since ANGULAR

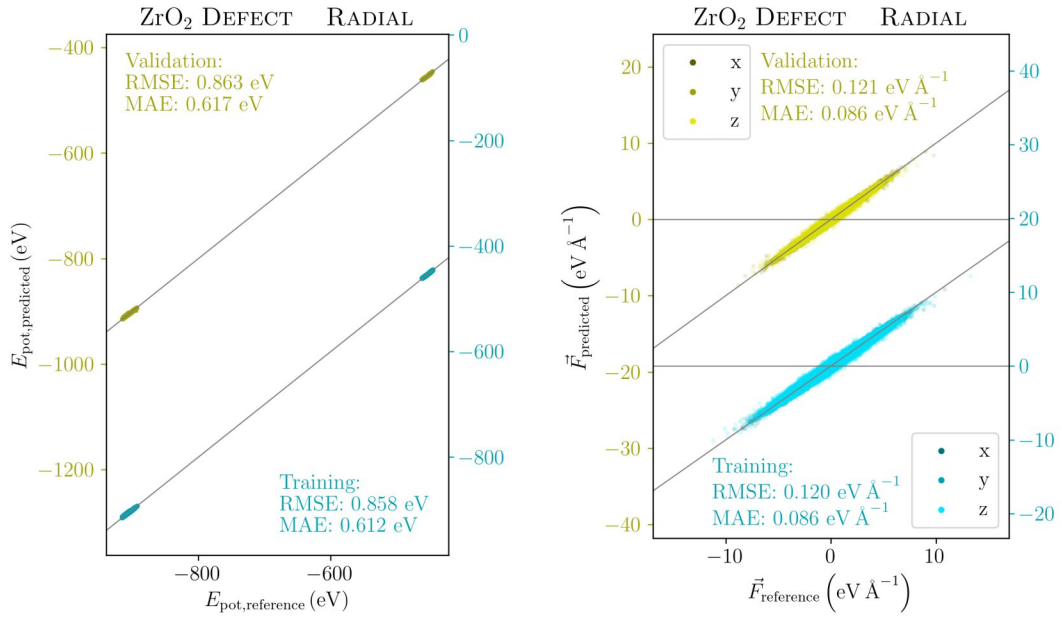


Figure 5.12: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO₂ DEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

are redundant (see chapter 4) this gives the NN an additional burden and can explain the high variability. Removing the redundancy with ODDANGULAR descriptors (also $n_{\text{max}}^{\mathcal{E}} = 8$) improves consistency but not the overall accuracy. Also the problem of the outlier BECs still persists.

Similar results can be seen for the ELEMENT descriptors. Large variability across training runs, but the best models also outperform models using RADIAL descriptors.

Using the different approach of EBESSEL (with $n_{\text{max}}^{\mathcal{E}} = n_{\text{max}} = 8$), where the electric field is introduced as a linear perturbation to the spherical Bessel descriptors, show consistent improvements in BEC prediction. This gain however, comes at the cost of much lower accuracy in the force predictions. The parity plot (figure 5.13, B.7) shows that outliers still remain, although fewer in numbers.

That the outliers remain is to some extent also related to the log-cosh loss, that weights large deviations less than MSE loss. Those ~ 1000 outliers may strongly deviate from the reference value, if fitting those points causes the overall RMSE to increase, the optimizer will effectively ignore them. To address this issue and include these outliers more in the training process, the MSE loss was used instead of the log-cosh loss. With the current log-cosh loss, this can be achieved by setting the log-cosh parameter excessively large and multiplying the loss by 2 (to account for the 0.5 MSE dependence of log-cosh for deviations smaller the scale parameter). The implementation in this thesis uses a BEC log-cosh parameter of $10e$ and a weight of $\lambda_{\text{BEC}} = 20$ instead of $\lambda_{\text{BEC}} = 10$. The models that were trained with the modified loss, EBESSEL* and ELEMENT* respectively.

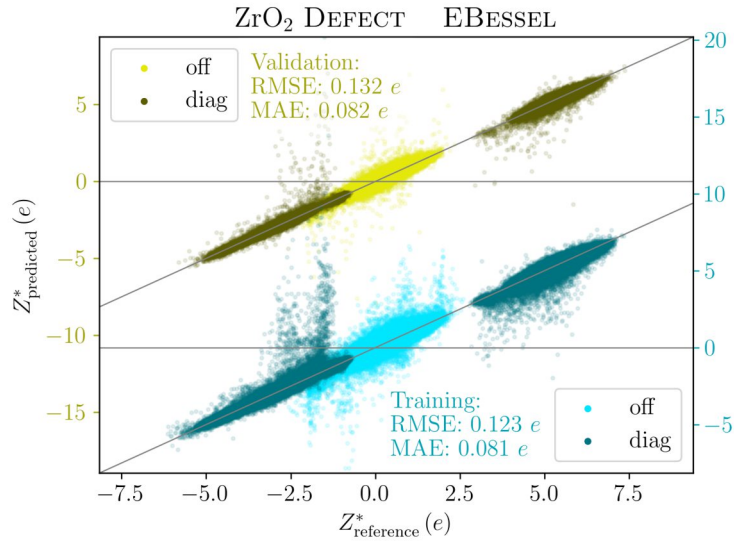


Figure 5.13: Prediction of BECs in the ZrO₂ DEFECT dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.

The modification of the loss proves to be effective. For EBESSEL* the outliers are reduced to ~ 300 and for ELEMENT* even down to ~ 150 . This can be appreciated in figure 5.14 for ELEMENT*. Not only the number but also the deviation of outliers is reduced, making this model much more appropriate for MD simulations under electric fields than models trained on the unmodified log-cosh loss. Moreover, the MSE loss puts less weight on small deviations of BEC, which has the effect that the model produces more accurate forces (see figure B.8), even though for the forces a non modified log-cosh loss was used.

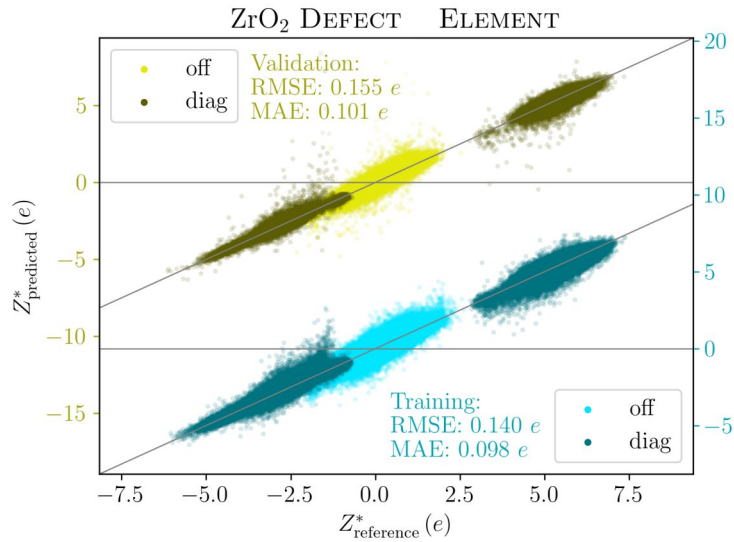


Figure 5.14: Prediction of BECs in the ZrO₂ DEFECT dataset using the unified differentiable learning of electric response framework with the ELEMENT descriptors. Trained with a BEC log cosh parameter of $10 e$ and $\lambda_{\text{BEC}} = 20$.

Li₃PO₄ Finally a set of 5000 pristine Li₃PO₄ structures was used for training. The parity plot of BEC prediction for a model using EBESSEL (with $n_{\max}^{\mathcal{E}} = n_{\max} = 6$) is shown in figure 5.15. The diagonal components can be decomposed into 3 distinct regions, corresponding from left to right (negative to positive charge) to oxygen, lithium and phosphorus. While the predictions of oxygen and phosphorus do not deviate considerably from the reference data, lithium displays considerable deviations, with some values being predicted close to zero or even with the wrong sign. However, those deviations are less pronounced than those of the ZrO₂ DEFECT dataset.

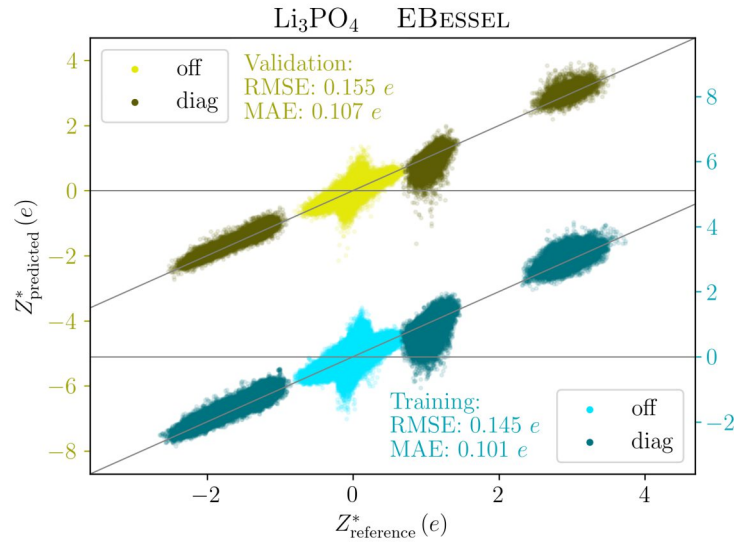


Figure 5.15: Prediction of BECs in the Li₃PO₄ dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.

To remedy the erroneous BEC prediction on the lithium atom, the BEC log-cosh parameter was again modified, so as to approximate the MSE loss. The result, shown in figure 5.17, indicate that even though points with the strongest deviation improved considerably, the parity plot remains largely unchanged.

Figure 5.16 shows the model comparison between EBESSEL and EBESSEL*. Similar to ZrO₂ DEFECT, switching to the MSE loss increases the final BEC RMSE, but more interestingly, the force RMSE approximately halves. This improvement is also visible in the parity plots for forces and energies (figures B.9 and B.10).

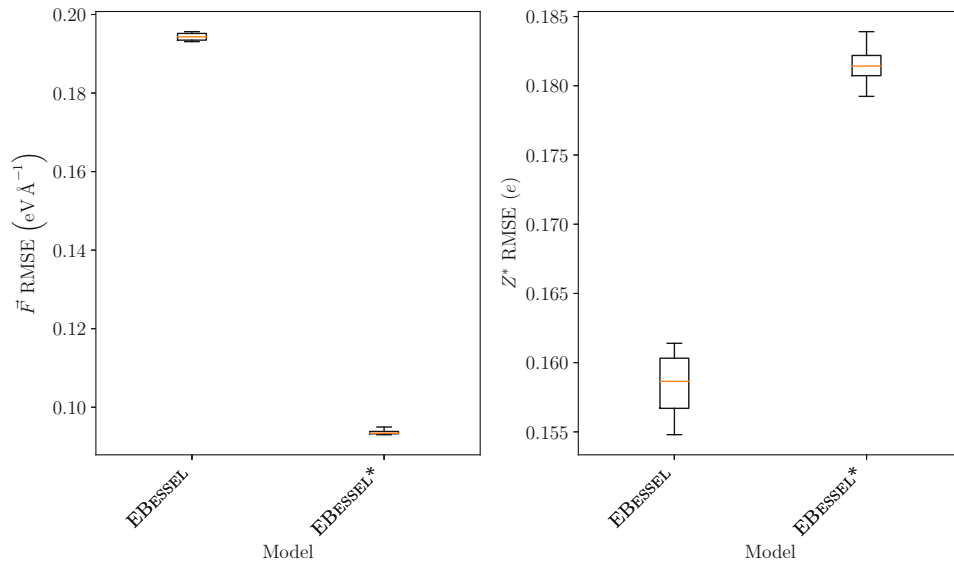


Figure 5.16: Final validation set performance of NEURALIL models on the Li_3PO_4 dataset. For each model, 4 repetitions were carried out. Models marked with * are trained with a logcosh parameter of $10e$ for the BEC and $\lambda_{\text{BEC}} = 20$.

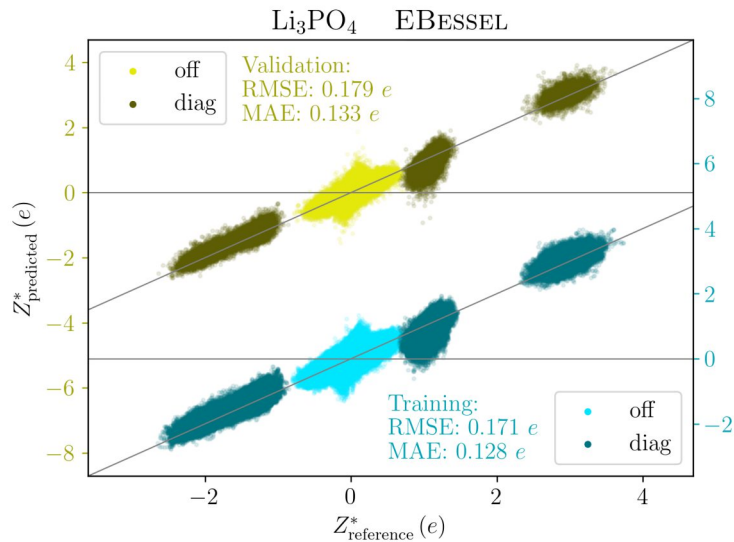


Figure 5.17: Prediction of BECs in the Li_3PO_4 dataset using theunified differentiable learning of electric response framework with the EBESSEL descriptors. Trained with a BEC logcosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$.

Chapter 6

Conclusion and Outlook

In this thesis, different approaches for predicting BEC using NNPs were explored. In an initial attempt, an unmodified NNP was used to predict forces induced by an external electric field, which is equivalent to fitting one column of the BEC tensor. This proved to be a poor choice: Applying an electric field breaks the symmetries, inducing forces that inherently cannot be fitted with a symmetry aware NNP.

To overcome this limitation, the unified differentiable learning of electric response from [16] was implemented in the NNP NEURALIL [38]. Obtaining the BECs as mixed derivative of the potential energy ensures that the predicted BECs satisfy exact physical constraints. The implementation required three main modifications: Modifying the loss function, adding the electric field vector as a model input and extending the derivatives of E_{pot} to higher order and mixed derivative with respect to positions and the electric field.

The main focus of this thesis was the development electric field descriptors. Since NEURALIL uses local rotation-invariant descriptors, the electric field descriptors have to reflect, how the electric field alters the symmetry. Simple descriptors based on dot products of interatomic distances and the electric field vector, denoted as RADIAL in this work, showed satisfactory accuracy for BEC predictions for simple datasets. However with larger and more diverse datasets, more powerful electric field descriptors were needed. Several different types descriptors were designed and tested in the scope of this thesis. The most accurate descriptors were those that add the electric field as a linear perturbation to the spherical Bessel descriptors (EBESSEL). When combining them with a modified loss function based on MSE, these descriptors had the overall best performance in providing both accurate BECs and forces, while avoiding outliers.

The unified differentiable learning of electric response framework can also predict polarizabilities and polarizations, in addition to the investigated potential energy, forces and BECs. These additional quantities were only investigated in the implementation phase for the water dimer dataset, but not when developing more accurate electric field descriptors. The main conclusions of this thesis therefore only apply to models that

predict the potential energy, forces and BECs at the same time, leaving the study of polarization and polarizability predictions for a future study.

Overall, the results demonstrate that it is possible to obtain tensorial quantities such as BEC using only rotation invariant descriptors. Moreover, both the electric field descriptors as well as the spherical Bessel descriptors are strictly local, with a receptive field of $\sim 5 \text{ \AA}$, allowing for very efficient predictions and parallelization across multiple compute devices. This enables us to perform large scale MD simulations under electric fields and deepen our understanding of electrochemical processes.

It would be of interest to not only assess BEC predictions with simple metrics such as RMSE or MAE, but also extend this to quantities obtained during MD simulations. Examples include diffusivities and ion conductivities, both of which have direct relevance for real-world devices such as batteries. Moreover, during MD simulations unseen configurations are likely to be encountered, allowing us to get a grasp of stability and transferability of the BEC predictions.

Bibliography

- [1] Roman M. Balabin and Ekaterina I. Lomakina. ‘Support vector machine regression (SVR/LS-SVM)—an alternative to neural networks (ANN) for analytical chemistry? Comparison of nonlinear methods on near infrared (NIR) spectroscopy data’. In: *The Analyst* 136.8 (2011), p. 1703. ISSN: 0003-2654, 1364-5528. DOI: 10.1039/c0an00387e. URL: <https://xlink.rsc.org/?DOI=c0an00387e> (visited on 08/06/2025).
- [2] Randall Balestriero, Jerome Pesenti and Yann LeCun. *Learning in High Dimension Always Amounts to Extrapolation*. 29th Oct. 2021. DOI: 10.48550/arXiv.2110.09485. arXiv: 2110.09485[cs]. URL: <http://arxiv.org/abs/2110.09485> (visited on 22/07/2025).
- [3] Albert P. Bartók, Risi Kondor and Gábor Csányi. ‘On representing chemical environments’. In: *Physical Review B* 87.18 (28th May 2013), p. 184115. ISSN: 1098-0121, 1550-235X. DOI: 10.1103/PhysRevB.87.184115. URL: <https://link.aps.org/doi/10.1103/PhysRevB.87.184115> (visited on 21/10/2024).
- [4] Albert P. Bartók et al. ‘Gaussian Approximation Potentials: the accuracy of quantum mechanics, without the electrons’. In: *Physical Review Letters* 104.13 (1st Apr. 2010), p. 136403. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.104.136403. arXiv: 0910.1019[physics]. URL: <http://arxiv.org/abs/0910.1019> (visited on 08/06/2025).
- [5] Ilyes Batatia et al. ‘MACE: Higher Order Equivariant Message Passing Neural Networks for Fast and Accurate Force Fields’. In: ().
- [6] Simon Batzner et al. ‘E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials’. In: *Nature Communications* 13.1 (4th May 2022), p. 2453. ISSN: 2041-1723. DOI: 10.1038/s41467-022-29939-5. URL: <https://www.nature.com/articles/s41467-022-29939-5> (visited on 12/11/2024).
- [7] Atilim Gunes Baydin et al. *Automatic differentiation in machine learning: a survey*. 5th Feb. 2018. DOI: 10.48550/arXiv.1502.05767. arXiv: 1502.05767[cs]. URL: <http://arxiv.org/abs/1502.05767> (visited on 22/07/2025).

- [8] Jörg Behler and Michele Parrinello. ‘Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces’. In: *Physical Review Letters* 98.14 (2nd Apr. 2007), p. 146401. ISSN: 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.98.146401. URL: <https://link.aps.org/doi/10.1103/PhysRevLett.98.146401>.
- [9] Thomas B. Blank et al. ‘Neural network models of potential energy surfaces’. In: *The Journal of Chemical Physics* 103.10 (8th Sept. 1995), pp. 4129–4137. ISSN: 0021-9606, 1089-7690. DOI: 10.1063/1.469597. URL: <https://pubs.aip.org/jcp/article/103/10/4129/481485/Neural-network-models-of-potential-energy-surfaces> (visited on 08/06/2025).
- [10] P. E. Blöchl. ‘Projector augmented-wave method’. In: *Physical Review B* 50.24 (15th Dec. 1994), pp. 17953–17979. ISSN: 0163-1829, 1095-3795. DOI: 10.1103/PhysRevB.50.17953. URL: <https://link.aps.org/doi/10.1103/PhysRevB.50.17953> (visited on 03/06/2025).
- [11] V. Botu et al. ‘Machine Learning Force Fields: Construction, Validation, and Outlook’. In: *The Journal of Physical Chemistry C* 121.1 (12th Jan. 2017), pp. 511–522. ISSN: 1932-7447, 1932-7455. DOI: 10.1021/acs.jpcc.6b10908. URL: <https://pubs.acs.org/doi/10.1021/acs.jpcc.6b10908> (visited on 22/08/2025).
- [12] Venkatesh Botu and Rampi Ramprasad. ‘Adaptive machine learning framework to accelerate *ab initio* molecular dynamics’. In: *International Journal of Quantum Chemistry* 115.16 (15th Aug. 2015), pp. 1074–1083. ISSN: 0020-7608, 1097-461X. DOI: 10.1002/qua.24836. URL: <https://onlinelibrary.wiley.com/doi/10.1002/qua.24836>.
- [13] James Bradbury et al. *JAX: composable transformations of Python+NumPy programs*. Version 0.3.13. 2018. URL: <http://github.com/jax-ml/jax>.
- [14] Jesús Carrete et al. ‘Deep Ensembles vs. Committees for Uncertainty Estimation in Neural-Network Force Fields: Comparison and Application to Active Learning’. In: *The Journal of Chemical Physics* 158.20 (28th May 2023). ISSN: 0021-9606, 1089-7690. DOI: 10.1063/5.0146905. arXiv: 2302.08805[physics]. URL: <http://arxiv.org/abs/2302.08805> (visited on 22/07/2025).
- [15] Wojciech Marian Czarnecki et al. *Sobolev Training for Neural Networks*. 26th July 2017. DOI: 10.48550/arXiv.1706.04859. arXiv: 1706.04859[cs]. URL: <http://arxiv.org/abs/1706.04859> (visited on 22/07/2025).
- [16] Stefano Falletta et al. *Unified Differentiable Learning of Electric Response*. 7th June 2024. arXiv: 2403.17207[cond-mat]. URL: <http://arxiv.org/abs/2403.17207> (visited on 16/10/2024).

- [17] Ph Ghosez, J. P. Michenaud and X. Gonze. ‘The physics of dynamical atomic charges: the case of ABO₃ compounds’. In: *Physical Review B* 58.10 (1st Sept. 1998), pp. 6224–6240. ISSN: 0163-1829, 1095-3795. DOI: 10.1103/PhysRevB.58.6224. arXiv: cond-mat/9805013. URL: <http://arxiv.org/abs/cond-mat/9805013> (visited on 26/04/2024).
- [18] Ph. Ghosez et al. ‘Born effective charges of barium titanate: Band-by-band decomposition and sensitivity to structural features’. In: *Physical Review B* 51.10 (1st Mar. 1995), pp. 6765–6768. ISSN: 0163-1829, 1095-3795. DOI: 10.1103/PhysRevB.51.6765. URL: <https://link.aps.org/doi/10.1103/PhysRevB.51.6765> (visited on 05/05/2024).
- [19] Xavier Gonze and Changyol Lee. ‘Dynamical matrices, Born effective charges, dielectric permittivity tensors, and interatomic force constants from density-functional perturbation theory’. In: *Physical Review B* 55.16 (15th Apr. 1997). Publisher: American Physical Society (APS), pp. 10355–10368. ISSN: 0163-1829, 1095-3795. DOI: 10.1103/physrevb.55.10355. URL: <https://link.aps.org/doi/10.1103/PhysRevB.55.10355> (visited on 22/07/2025).
- [20] B. Hammer, L. B. Hansen and J. K. Nørskov. ‘Improved adsorption energetics within density-functional theory using revised Perdew-Burke-Ernzerhof functionals’. In: *Physical Review B* 59.11 (15th Mar. 1999), pp. 7413–7421. ISSN: 0163-1829, 1095-3795. DOI: 10.1103/PhysRevB.59.7413. URL: <https://link.aps.org/doi/10.1103/PhysRevB.59.7413> (visited on 04/06/2025).
- [21] Kaiming He et al. *Deep Residual Learning for Image Recognition*. 10th Dec. 2015. DOI: 10.48550/arXiv.1512.03385. arXiv: 1512.03385[cs]. URL: <http://arxiv.org/abs/1512.03385> (visited on 22/07/2025).
- [22] Graeme Henkelman, Andri Arnaldsson and Hannes Jónsson. ‘A fast and robust algorithm for Bader decomposition of charge density’. In: *Computational Materials Science* 36.3 (June 2006), pp. 354–360. ISSN: 09270256. DOI: 10.1016/j.commatsci.2005.04.010. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0927025605001849> (visited on 05/06/2025).
- [23] P. Hohenberg and W. Kohn. ‘Inhomogeneous Electron Gas’. In: *Physical Review* 136.3 (9th Nov. 1964). Publisher: American Physical Society (APS), B864–B871. ISSN: 0031-899X. DOI: 10.1103/physrev.136.b864. URL: <https://link.aps.org/doi/10.1103/PhysRev.136.B864> (visited on 22/07/2025).
- [24] Kurt Hornik, Maxwell Stinchcombe and Halbert White. ‘Multilayer feedforward networks are universal approximators’. In: *Neural Networks* 2.5 (Jan. 1989), pp. 359–366. ISSN: 08936080. DOI: 10.1016/0893-6080(89)90020-8. URL: <https://linkinghub.elsevier.com/retrieve/pii/0893608089900208> (visited on 08/06/2025).

- [25] Weihua Hu et al. *ForceNet: A Graph Neural Network for Large-Scale Quantum Calculations*. 2nd Mar. 2021. DOI: 10.48550/arXiv.2103.01436. arXiv: 2103.01436[cs]. URL: <http://arxiv.org/abs/2103.01436> (visited on 28/05/2025).
- [26] William L. Jorgensen, David S. Maxwell and Julian Tirado-Rives. ‘Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids’. In: *Journal of the American Chemical Society* 118.45 (13th Nov. 1996), pp. 11225–11236. ISSN: 0002-7863, 1520-5126. DOI: 10.1021/ja9621760. URL: <https://pubs.acs.org/doi/10.1021/ja9621760> (visited on 22/08/2025).
- [27] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 30th Jan. 2017. DOI: 10.48550/arXiv.1412.6980. arXiv: 1412.6980[cs]. URL: <http://arxiv.org/abs/1412.6980> (visited on 29/07/2025).
- [28] Emir Kocer, Jeremy K. Mason and Hakan Erturk. ‘A novel approach to describe chemical environments in high-dimensional neural network potentials’. In: *The Journal of Chemical Physics* 150.15 (21st Apr. 2019), p. 154102. ISSN: 0021-9606, 1089-7690. DOI: 10.1063/1.5086167. URL: <https://pubs.aip.org/jcp/article/150/15/154102/76113/A-novel-approach-to-describe-chemical-environments> (visited on 09/06/2025).
- [29] Emir Kocer, Jeremy K. Mason and Hakan Erturk. ‘Continuous and optimally complete description of chemical environments using Spherical Bessel descriptors’. In: *AIP Advances* 10.1 (1st Jan. 2020), p. 015021. ISSN: 2158-3226. DOI: 10.1063/1.5111045. URL: <https://pubs.aip.org/adv/article/10/1/015021/1036876/Continuous-and-optimally-complete-description-of>.
- [30] W. Kohn and L. J. Sham. ‘Self-Consistent Equations Including Exchange and Correlation Effects’. In: *Physical Review* 140.4 (15th Nov. 1965). Publisher: American Physical Society (APS), A1133–A1138. ISSN: 0031-899X. DOI: 10.1103/physrev.140.a1133. URL: <https://link.aps.org/doi/10.1103/PhysRev.140.A1133> (visited on 22/07/2025).
- [31] G. Kresse and J. Furthmüller. ‘Efficiency of ab-initio total energy calculations for metals and semiconductors using a plane-wave basis set’. In: *Computational Materials Science* 6.1 (July 1996), pp. 15–50. ISSN: 09270256. DOI: 10.1016/0927-0256(96)00008-0. URL: <https://linkinghub.elsevier.com/retrieve/pii/0927025696000080> (visited on 04/06/2025).
- [32] G. Kresse and J. Furthmüller. ‘Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set’. In: *Physical Review B* 54.16 (15th Oct. 1996), pp. 11169–11186. ISSN: 0163-1829, 1095-3795. DOI: 10.1103/PhysRevB.54.11169. URL: <https://link.aps.org/doi/10.1103/PhysRevB.54.11169> (visited on 04/06/2025).

- [33] Alex Kutana et al. ‘Representing Born effective charges with equivariant graph convolutional neural networks’. In: *Scientific Reports* 15.1 (14th May 2025), p. 16719. ISSN: 2045-2322. DOI: 10.1038/s41598-025-01250-5. URL: <https://www.nature.com/articles/s41598-025-01250-5> (visited on 27/05/2025).
- [34] Naoaki Kuwata et al. ‘Characterization of Thin-Film Lithium Batteries with Stable Thin-Film Li₃PO₄ Solid Electrolytes Fabricated by ArF Excimer Laser Deposition’. In: *Journal of The Electrochemical Society* (11th Mar. 2010).
- [35] Sönke Lorenz, Axel Groß and Matthias Scheffler. ‘Representing high-dimensional potential-energy surfaces for reactions at surfaces by neural networks’. In: *Chemical Physics Letters* 395.4 (Sept. 2004), pp. 210–215. ISSN: 00092614. DOI: 10.1016/j.cplett.2004.07.076. URL: <https://linkinghub.elsevier.com/retrieve/pii/S000926140401125X> (visited on 08/06/2025).
- [36] Luke Metz et al. ‘Practical tradeoffs between memory, compute, and performance in learned optimizers’. In: *Conference on Lifelong Learning Agents (CoLLAs)*. 2022. URL: http://github.com/google/learned_optimization.
- [37] Luke Metz et al. *VeLO: Training Versatile Learned Optimizers by Scaling Up*. 17th Nov. 2022. DOI: 10.48550/arXiv.2211.09760. arXiv: 2211.09760[cs]. URL: <http://arxiv.org/abs/2211.09760> (visited on 22/07/2025).
- [38] Hadrián Montes-Campos et al. ‘A Differentiable Neural-Network Force Field for Ionic Liquids’. In: *Journal of Chemical Information and Modeling* 62.1 (10th Jan. 2022). Publisher: American Chemical Society (ACS), pp. 88–101. ISSN: 1549-9596, 1549-960X. DOI: 10.1021/acs.jcim.1c01380. URL: <https://pubs.acs.org/doi/10.1021/acs.jcim.1c01380> (visited on 22/07/2025).
- [39] R. S. Mulliken. ‘Electronic Population Analysis on LCAO-MO Molecular Wave Functions. I’. In: *The Journal of Chemical Physics* 23.10 (1st Oct. 1955), pp. 1833–1840. ISSN: 0021-9606, 1089-7690. DOI: 10.1063/1.1740588. URL: <https://pubs.aip.org/jcp/article/23/10/1833/76789/Electronic-Population-Analysis-on-LCAO-MO> (visited on 05/06/2025).
- [40] Albert Musaelian et al. ‘Learning local equivariant representations for large-scale atomistic dynamics’. In: *Nature Communications* 14.1 (3rd Feb. 2023), p. 579. ISSN: 2041-1723. DOI: 10.1038/s41467-023-36329-y. URL: <https://www.nature.com/articles/s41467-023-36329-y> (visited on 08/04/2024).
- [41] Prajit Ramachandran, Barret Zoph and Quoc V. Le. *Searching for Activation Functions*. 27th Oct. 2017. DOI: 10.48550/arXiv.1710.05941. arXiv: 1710.05941[cs]. URL: <http://arxiv.org/abs/1710.05941> (visited on 07/07/2025).
- [42] Johannes Schorghuber. ‘Electrostatic Interactions in Neural-Network Force Fields’. Masterthesis. Vienna: TU Wien, 9th May 2023. 48 pp. URL: <https://repositum.tuwien.at/handle/20.500.12708/177637>.

- [43] Kristof T. Schütt et al. *SchNet: A continuous-filter convolutional neural network for modeling quantum interactions*. 19th Dec. 2017. arXiv: 1706.08566[stat]. URL: <http://arxiv.org/abs/1706.08566> (visited on 12/11/2024).
- [44] Thomas P. Senftle et al. ‘The ReaxFF reactive force-field: development, applications and future directions’. In: *npj Computational Materials* 2.1 (4th Mar. 2016), p. 15011. ISSN: 2057-3960. DOI: 10.1038/npjcompumats.2015.11. URL: <https://www.nature.com/articles/npjcompumats201511> (visited on 22/08/2025).
- [45] Alexander V. Shapeev. ‘Moment Tensor Potentials: A Class of Systematically Improvable Interatomic Potentials’. In: *Multiscale Modeling & Simulation* 14.3 (Jan. 2016), pp. 1153–1173. ISSN: 1540-3459, 1540-3467. DOI: 10.1137/15M1054183. URL: <http://epubs.siam.org/doi/10.1137/15M1054183> (visited on 09/06/2025).
- [46] Koji Shimizu et al. *Prediction of Born effective charges using neural network to study ion migration under electric fields: applications to crystalline and amorphous Li₃PO₄*. 31st May 2023. arXiv: 2305.19546[cond-mat]. URL: <http://arxiv.org/abs/2305.19546>.
- [47] J. S. Smith, O. Isayev and A. E. Roitberg. ‘ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost’. In: *Chemical Science* 8.4 (2017), pp. 3192–3203. ISSN: 2041-6520, 2041-6539. DOI: 10.1039/C6SC05720A. URL: <https://xlink.rsc.org/?DOI=C6SC05720A> (visited on 08/06/2025).
- [48] Nicola A. Spaldin. ‘A beginner’s guide to the modern theory of polarization’. In: *Journal of Solid State Chemistry* 195 (Nov. 2012), pp. 2–10. ISSN: 00224596. DOI: 10.1016/j.jssc.2012.05.010. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0022459612003234>.
- [49] S. Sra, S. Nowozin and S.J. Wright. *Optimization for Machine Learning*. Neural information processing series. MIT Press, 2012. ISBN: 9780262016469. URL: <https://books.google.at/books?id=JPQx7s2L1A8C>.
- [50] Nathaniel Thomas et al. *Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds*. 18th May 2018. arXiv: 1802.08219[cs]. URL: <http://arxiv.org/abs/1802.08219> (visited on 12/11/2024).
- [51] Yaolong Zhang and Bin Jiang. ‘Universal machine learning for the response of atomistic systems to external fields’. In: *Nature Communications* 14.1 (12th Oct. 2023), p. 6424. ISSN: 2041-1723. DOI: 10.1038/s41467-023-42148-y. URL: <https://www.nature.com/articles/s41467-023-42148-y> (visited on 16/10/2024).

List of abbreviations

AIMD *Ab initio* molecular dynamics. xii, 1, 19, 37, 54

BEC Born effective charge. xi–xiii, 1, 2, 8, 13–15, 17–21, 26, 27, 29, 31–44, 53, 54, 61, 62

BP-SF Behler-Parinello symmetry functions used in [8] to describe the atomic environment. 6

CNN Convolutional neural network. 4, 10

DFPT Density functional perturbation theory [19]. 14, 17, 32

DFT Density functional theory. Here Kohn-Sham density functional theory is assumed [23, 30]. 14

E(3) Euclidean group consisting of all translations, rotations and reflections in three dimensional space. 5, 6, 8, 10, 11, 21, 31

FFNN Feed-forward neural network. 4, 5, 8, 10, 15

GCNN Graph convolutional neural network. 4, 10, 11, 20

GNN Graph neural network. 15

MAE Mean average error. 9

MD Molecular dynamics. 1, 2, 18, 37, 38, 40, 44

ML Machine learning. 1–3, 5, 10, 11

MLFF Machine learning force field. 1–3, 14

MLP Machine learning lotential. 6, 10

MSE Mean square error. 9, 39–41, 43

NN Neural network. 3–6, 9–11, 15, 21, 23, 25, 27, 33, 39

NNFF Neural network force field. 19, 29–31

NNP Neural network potential. xi, xii, 2, 6, 8, 10, 14, 15, 25, 33, 34, 43

PAW Projector augmented wave. 17

PES Potential energy surface. 3, 5, 9, 10

RMSE Root mean square error. 20, 26, 37, 39, 41

VASP Vienna Ab-initio Simulation Package. xi, xii, 17–19, 53, 54

Appendix A

Methodology

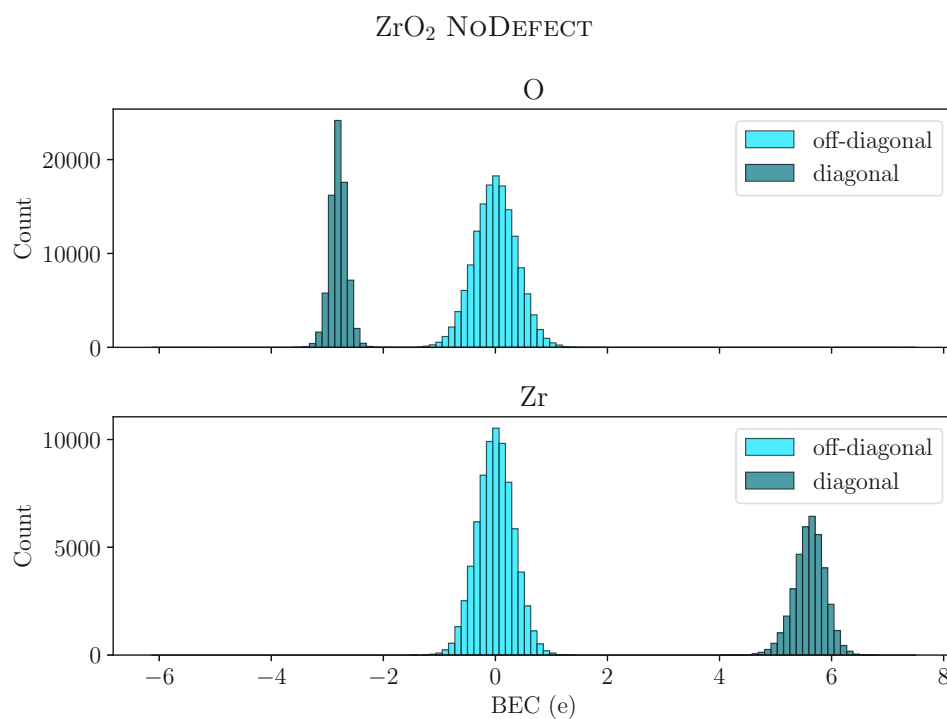


Figure A.1: Histogram showing the distribution of BECs calculated in VASP for ZrO₂ NoDEFECT

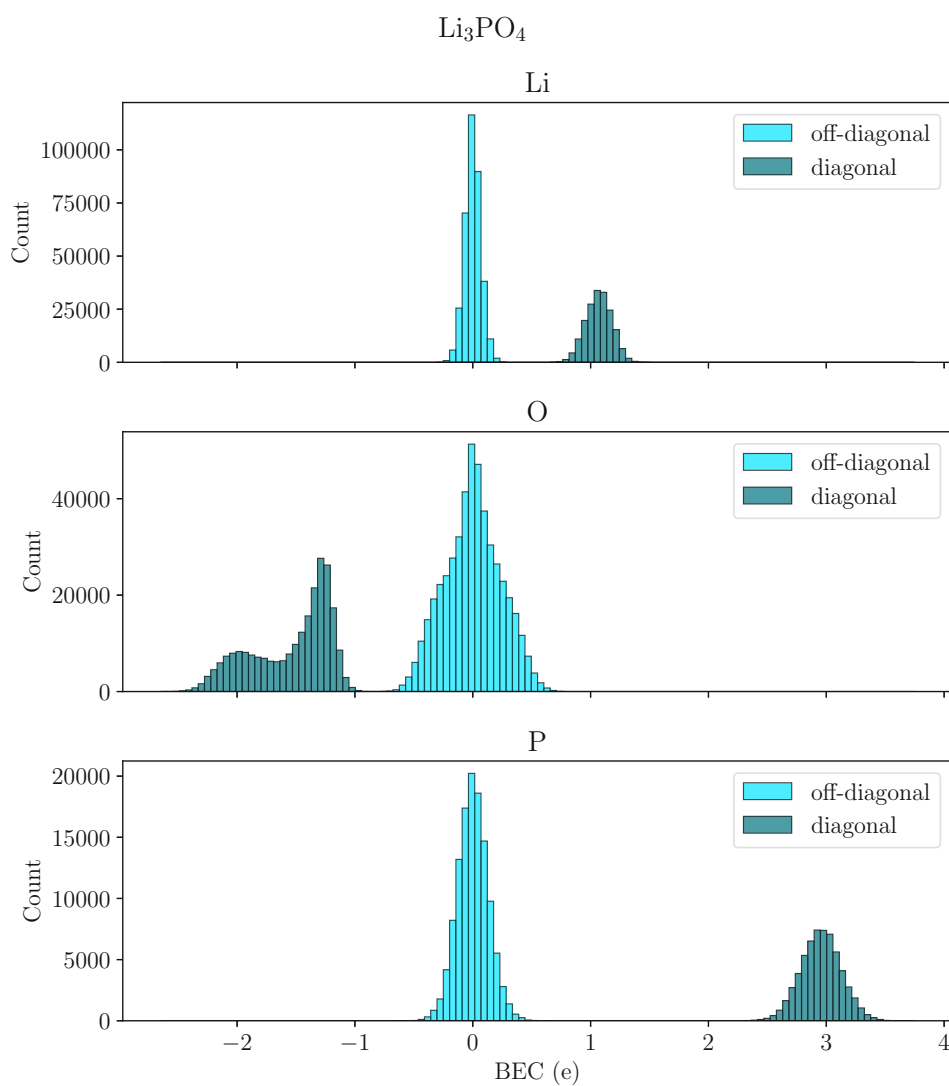


Figure A.2: Histogram showing the distribution of BECs calculated in VASP for a dataset of pristine Li_3PO_4 generated with AIMD at 2000 K.

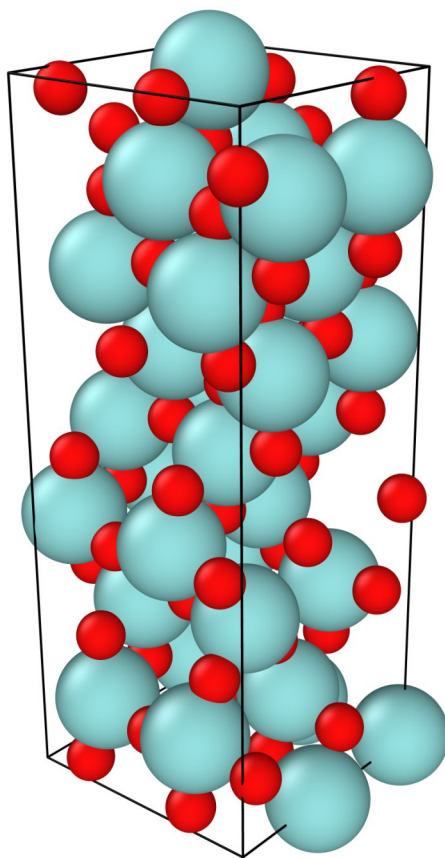
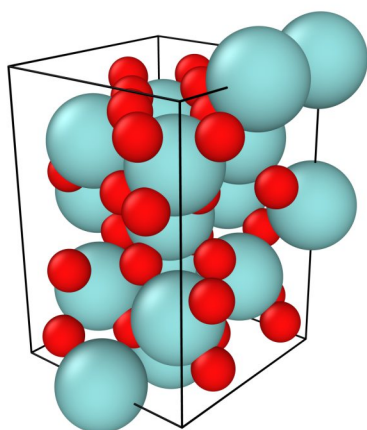
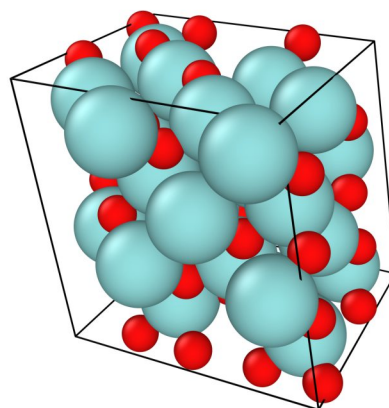


Figure A.3: One tetragonal structure (mp-754403) from ZrO₂ DEFECT.



(a) Tetragonal



(b) Monoclinic

Figure A.4: (a) One tetragonal structure (mp-2574) and (b) one monoclinic structure (mp-2858) from ZrO₂ DEFECT.

Appendix B

Results

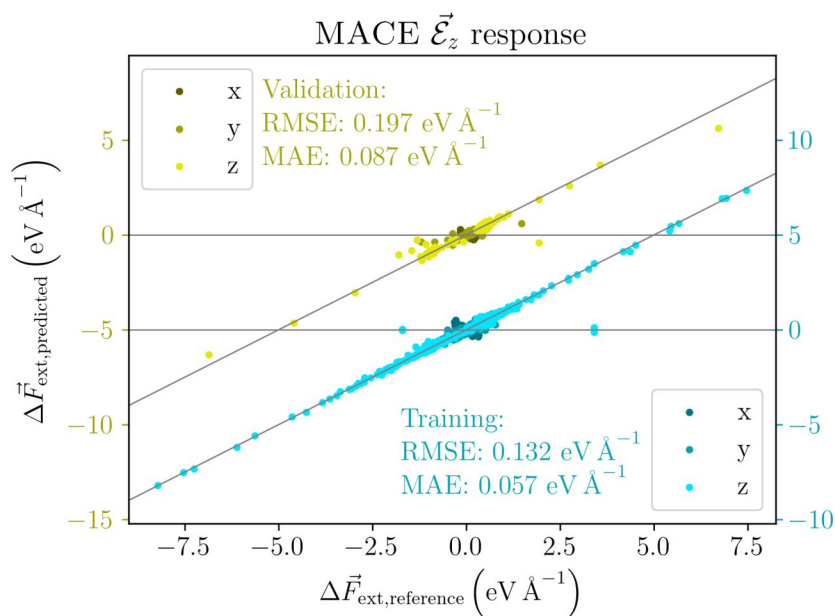


Figure B.1: Fitting of forces $\Delta \vec{F}_{\text{ext}}$ induced by the electric field $\vec{\mathcal{E}}_z$ using MACE.

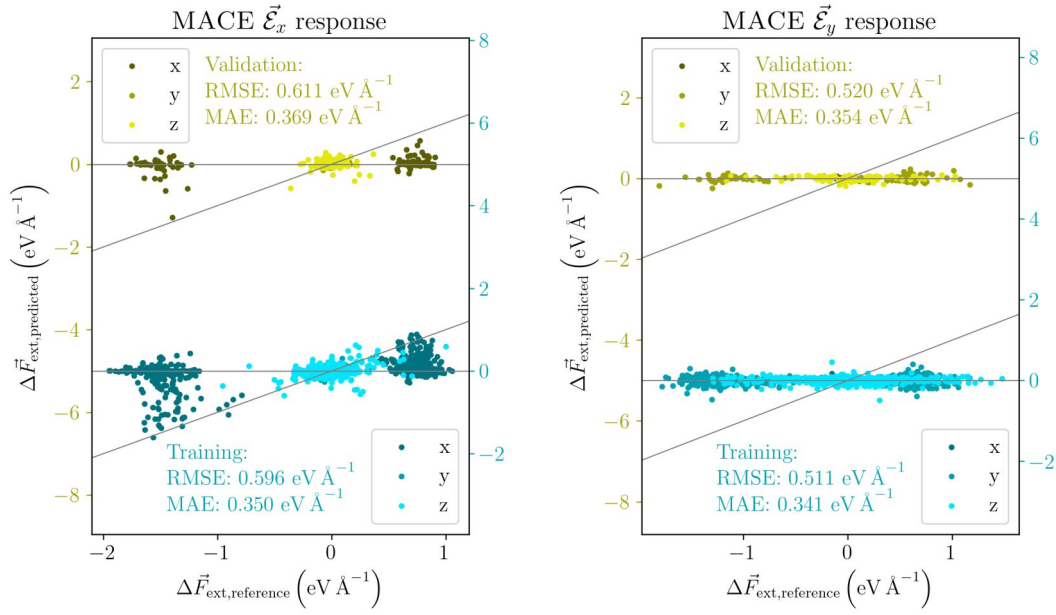


Figure B.2: Fitting of forces $\Delta \vec{F}_{\text{ext}}$ induced by electric fields $\vec{E}_x$ and $\vec{E}_y$ using MACE.

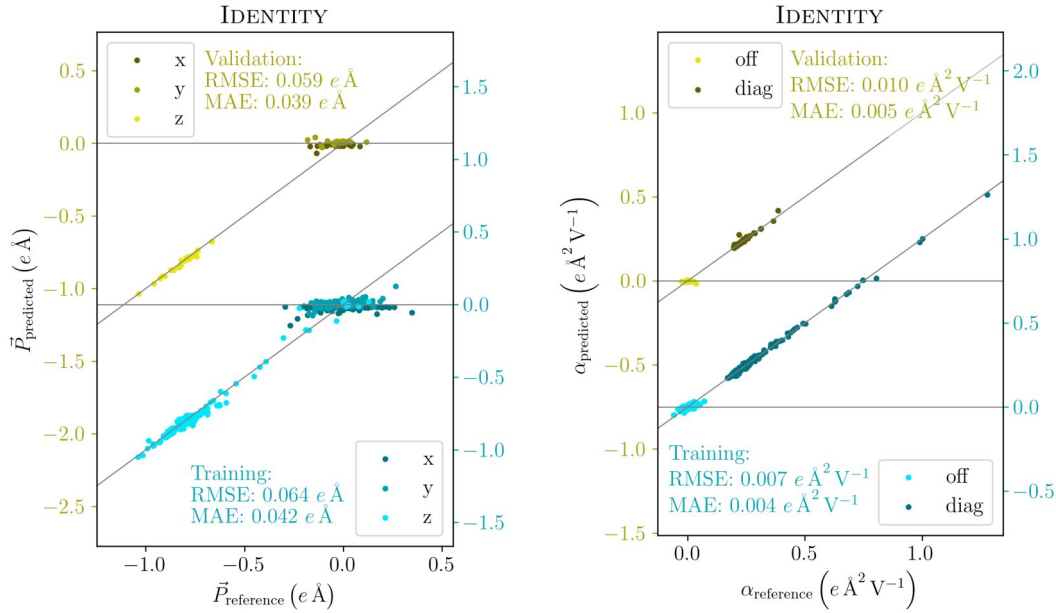


Figure B.3: Prediction of polarizations $\vec{P}$ and polarizabilities α in the water dimer dataset using the unified differentiable learning of electric response framework with the IDENTITY descriptors.

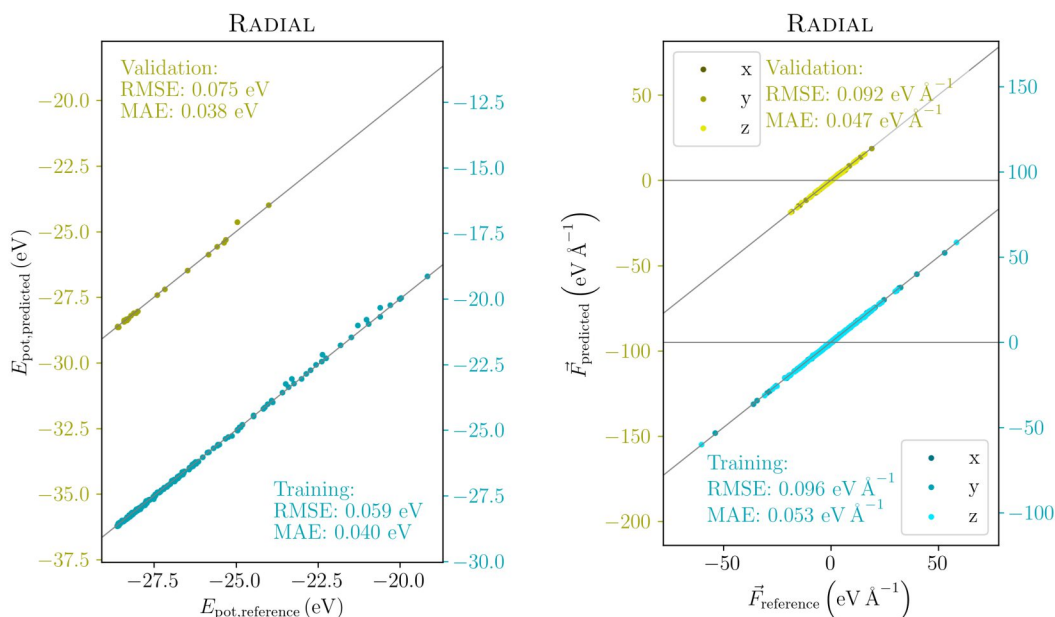


Figure B.4: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the water dimer dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

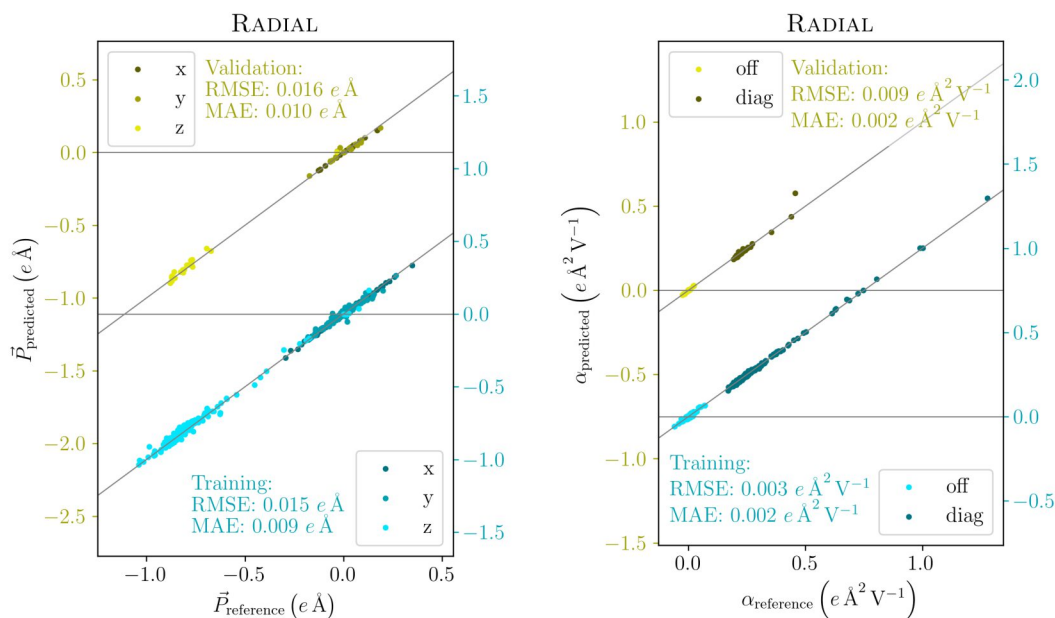


Figure B.5: Prediction of polarizations $\vec{P}$ and polarizabilities α in the water dimer dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

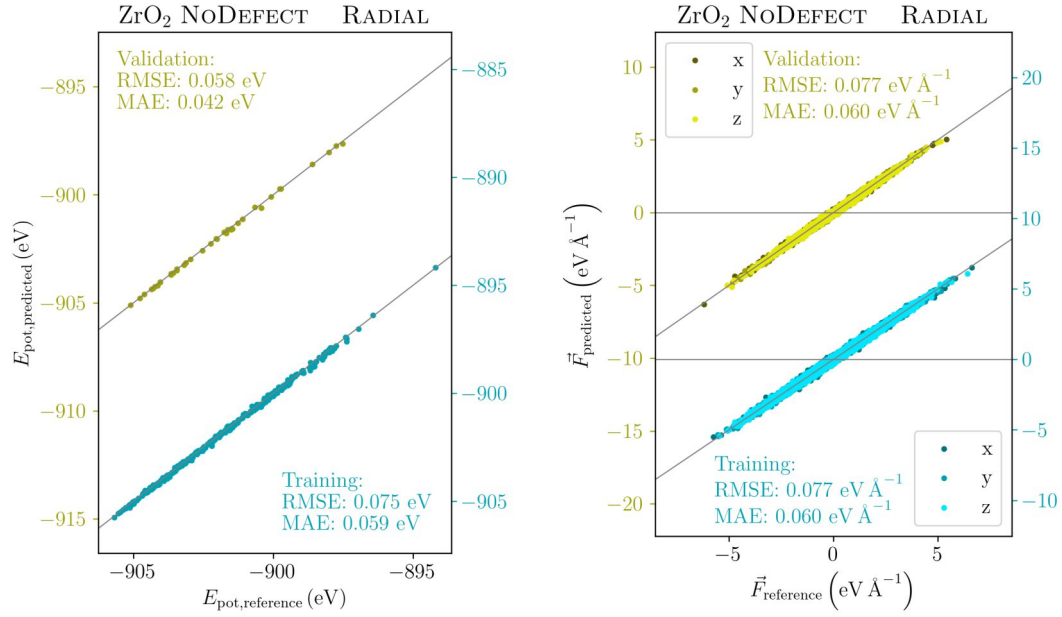


Figure B.6: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO₂ NoDEFECT dataset using the unified differentiable learning of electric response framework with the RADIAL descriptors.

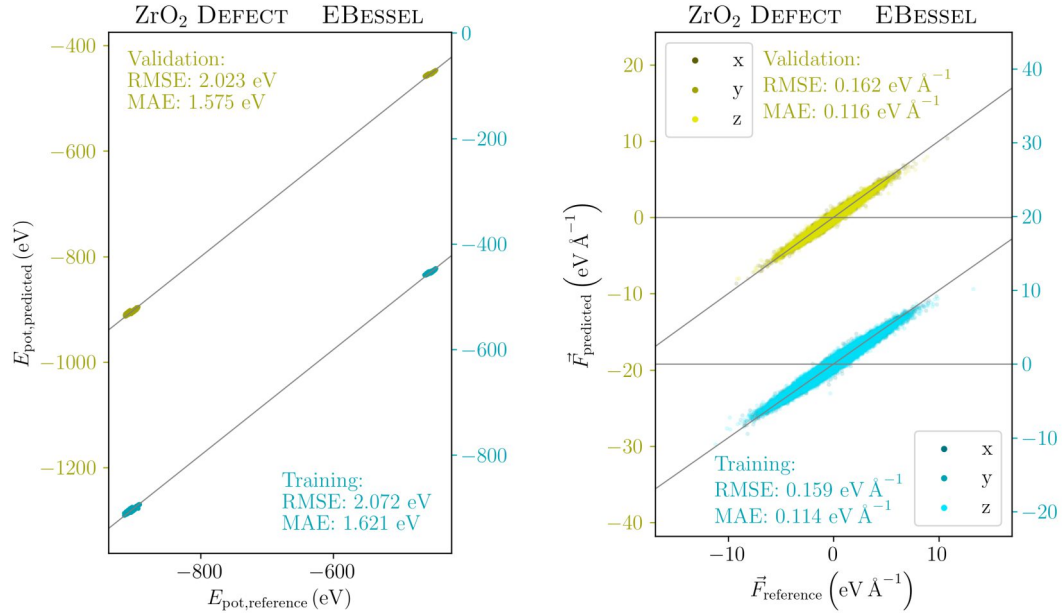


Figure B.7: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO₂ DEFECT dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.

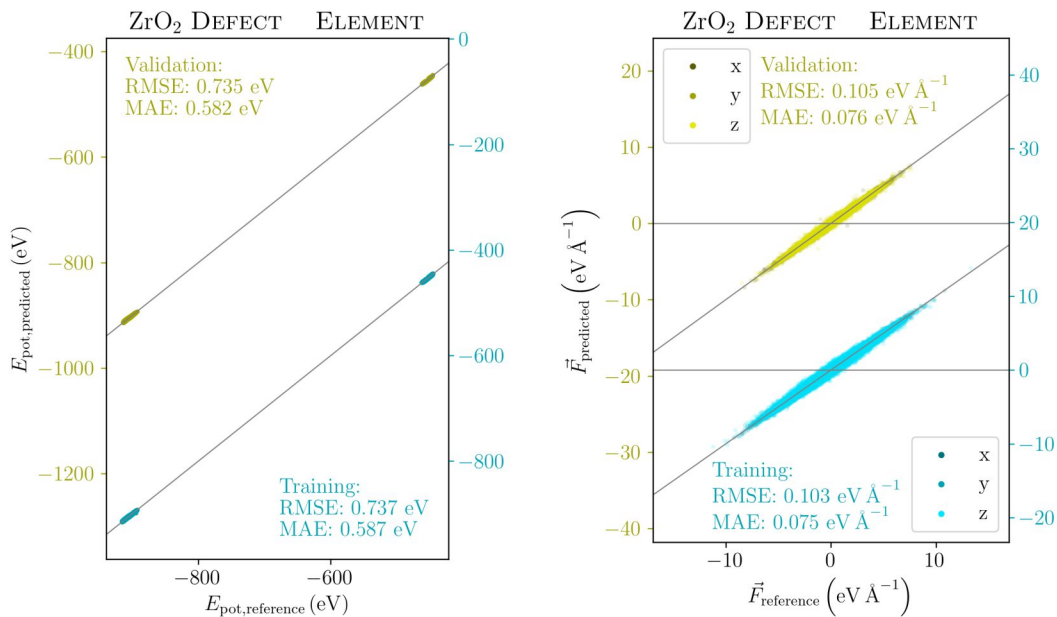


Figure B.8: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the ZrO_2 DEFECT dataset using the unified differentiable learning of electric response framework with the ELEMENT descriptors. Trained with a BEC cosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$.

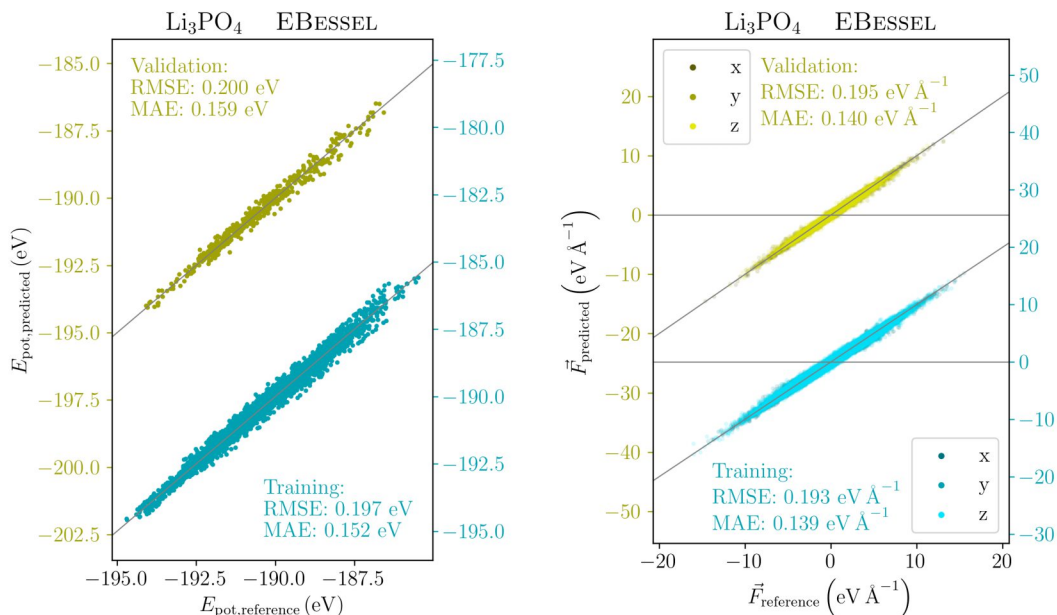


Figure B.9: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the Li_3PO_4 dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors.

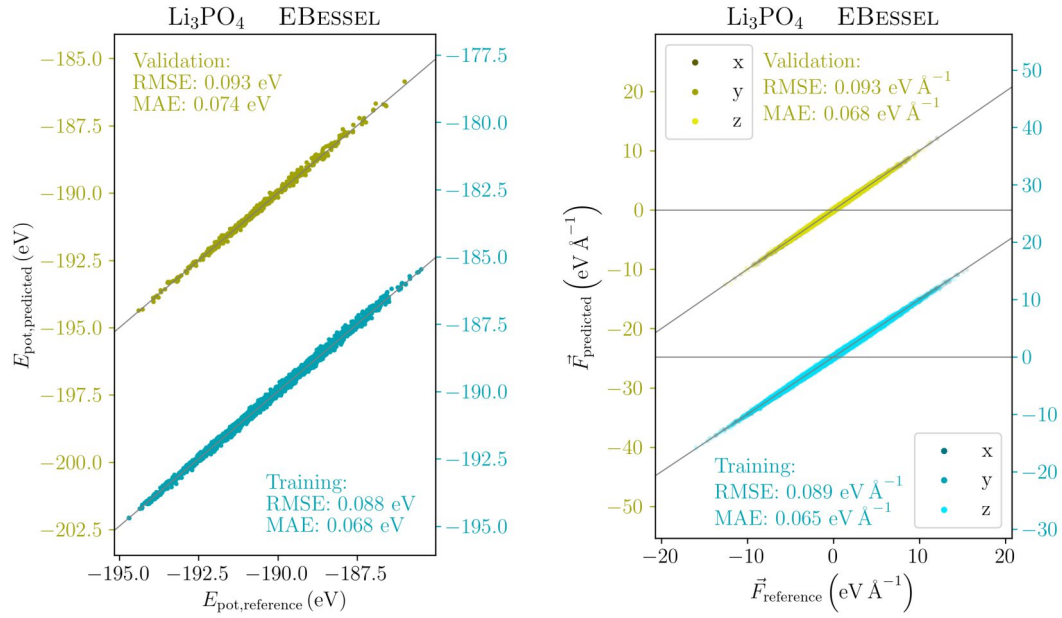


Figure B.10: Prediction of the potential energy E_{pot} and forces $\vec{F}$ in the Li_3PO_4 dataset using the unified differentiable learning of electric response framework with the EBESSEL descriptors. Trained with a BEC logcosh parameter of $10e$ and $\lambda_{\text{BEC}} = 20$.