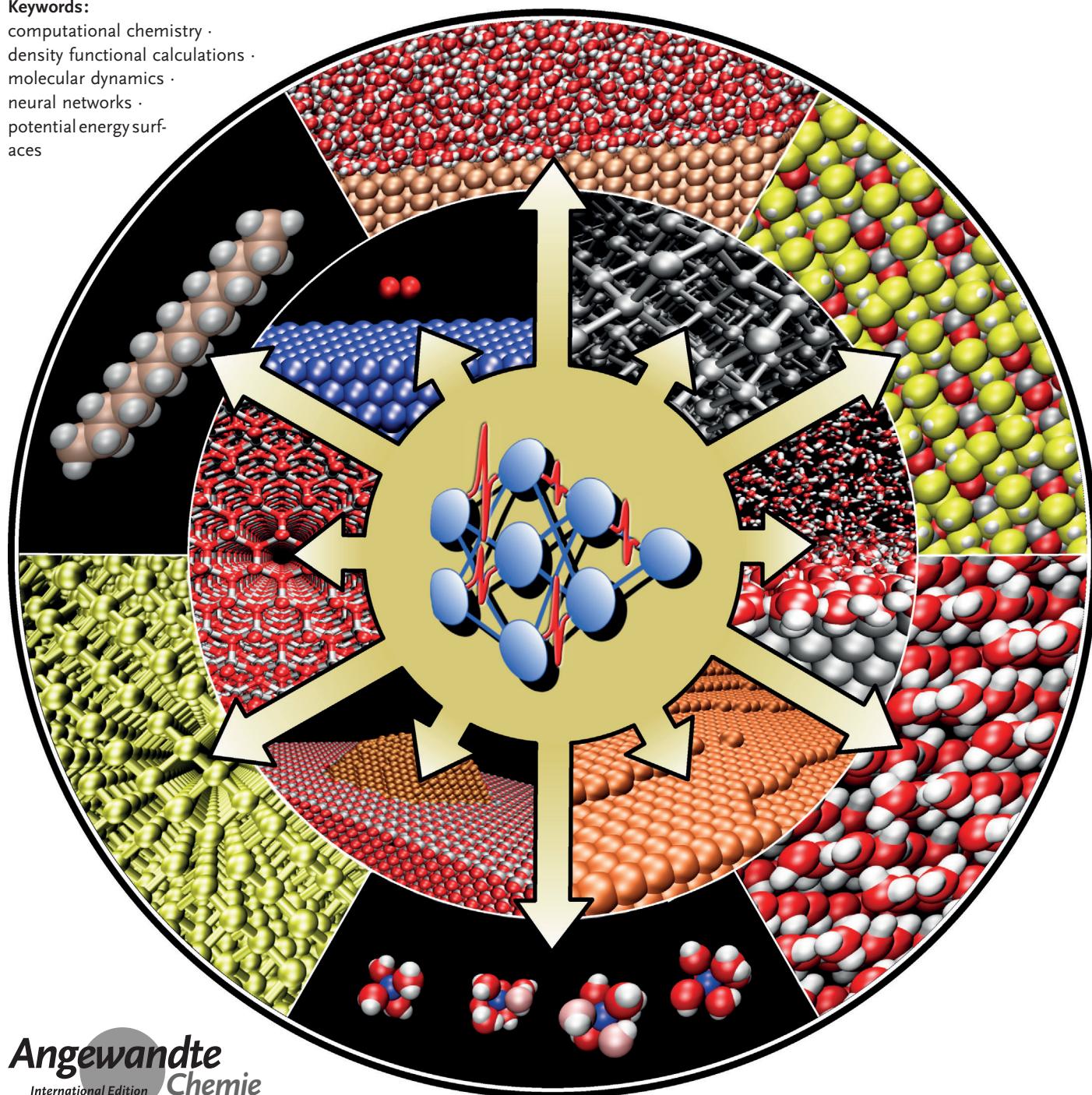


First Principles Neural Network Potentials for Reactive Simulations of Large Molecular and Condensed Systems

Jörg Behler*

Keywords:

computational chemistry ·
density functional calculations ·
molecular dynamics ·
neural networks ·
potential energy surfaces



Modern simulation techniques have reached a level of maturity which allows a wide range of problems in chemistry and materials science to be addressed. Unfortunately, the application of first principles methods with predictive power is still limited to rather small systems, and despite the rapid evolution of computer hardware no fundamental change in this situation can be expected. Consequently, the development of more efficient but equally reliable atomistic potentials to reach an atomic level understanding of complex systems has received considerable attention in recent years. A promising new development has been the introduction of machine learning (ML) methods to describe the atomic interactions. Once trained with electronic structure data, ML potentials can accelerate computer simulations by several orders of magnitude, while preserving quantum mechanical accuracy. This Review considers the methodology of an important class of ML potentials that employs artificial neural networks.

1. Introduction

Advances in science and technology rely on new materials and molecules with tailored chemical, mechanical, electronic, and optical properties. It is now the consensus that it is impossible to meet these requirements by using conventional trial and error approaches. Instead, the design of novel molecules and materials must be based on a fundamental understanding of the relationship between their properties and the atomistic as well as the electronic structure, their dynamical characteristics, and their reactivity, which are all intimately related to the atomic and molecular interactions. Deeper physical and chemical insights are important for applications as diverse as the development of alloys that combine mechanical strength and corrosion resistance with reduced weight, the identification of selective drugs, the improvement of catalysts for the chemical industry, biocompatible materials for medical implants, fail-safe nanoscale electronic devices, efficient battery technologies for electrochemical energy storage and conversion, as well as for functionalized surfaces and hybrid materials. Unraveling the underlying principles is crucial, as even small changes, for example, in composition, can have dramatic effects on the properties of interest.

In all these fields, rapid progress in experimental techniques has led to fascinating new insights. However, the interpretation of this wealth of experimental data is impossible without complementary information from computer simulations, and there is no hope for a comprehensive understanding at the atomic level without a joint effort of experiment and theory. Thus, despite modern techniques, which offer atomic resolution, such as scanning tunneling microscopy, transmission electron microscopy, and atom probe tomography, and advanced spectroscopic methods the importance of theoretical studies is constantly growing. As a consequence, a lot of effort is spent on the development of theoretical tools that are applicable to systems of increasing complexity, with the final aim of achieving predictive power.

From the Contents

1. Introduction	12829
2. Machine Learning Potentials	12831
3. Neural Network Potentials	12832
4. Discussion	12836
5. Summary and Outlook	12838

To reach this goal, the accuracy of first principles methods is required. Despite a few well-known shortcomings of current approximate exchange correlation functionals,^[1] density functional theory (DFT)^[2] offers a good compromise between efficiency and accuracy for many systems. Accordingly, DFT has gained a central role in the theoretical investigation of condensed systems containing up to about one thousand atoms. In particular, ab initio molecular dynamics (MD) simulations rely almost exclusively on DFT.^[3,4] For moderately sized molecules, on the other hand, a hierarchy of very accurate wave-function-based methods with excellent error control is available,^[5] which are often on a par with the best experimental data. Both DFT and wave function methods have been tremendously successful in contributing to the understanding of many problems in chemistry, physics, and materials science.

Although electronic structure calculations have become comparably cheap on modern supercomputers and are routinely applied in many fields, theoreticians face a dilemma, as the complexity of the systems of interest has also grown dramatically. To keep up with current needs it is, for example, necessary to abandon idealized model systems, to search a wider range of materials, structures, and compositions, to follow the evolution of systems over substantially longer time scales, and to resolve even subtle details of the atomic interactions, such as reaction barriers, with high precision.

The ability to calculate first-principles data for thousands of structures is without doubt a big step forward in achieving these goals, but it is not a general solution, because the analysis of the resulting large amounts of data might easily become the new bottleneck, thereby hindering a thorough understanding. Furthermore, for numerous problems first principles methods are, and will remain, many orders of magnitude too demanding. Consequently, there is no hope

[*] Prof. Dr. J. Behler
Universität Göttingen
Institut für Physikalische Chemie, Theoretische Chemie
Tammannstrasse 6, 37077 Göttingen (Germany)
E-mail: joerg.behler@uni-goettingen.de

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/anie.201703114>.

that it will just be a matter of time and progress in computer technology until a direct application of first principles methods will allow arbitrary problems to be solved. The costs of using realistic structural models containing thousands or even millions of atoms, of considering a huge number of configurations in long simulations, or of dealing with the exponentially growing number of possible materials in high-throughput screenings will remain prohibitive.

This need for larger systems and longer simulations is not new and has challenged theoreticians for generations. Different strategies have been devised for different types of problems, and satisfying solutions have been found in several fields, such as the development of classical force fields for studying the dynamics of large biomolecules in solution^[6–8] or QM/MM methods for localized chemical reactions in a large matrix.^[9] What is new is that other fields which have traditionally been the domain of electronic structure methods have entered the stage in recent years. Among these are problems in materials science, reactions in aqueous solution involving the dissociation and recombination of water molecules, and chemical processes at interfaces. All these systems require a reliable description of bond breaking and bond making as well as of complex bonding situations, atomic environments, and structural rearrangements. All this is naturally taken into account in first principles calculations, while on the other hand there is still a lack of efficient atomistic potentials for these applications.

Thus, to avoid electronic structure calculations for such systems, improved atomistic potentials are needed which provide a direct functional relationship between the atomic positions and the potential energy, namely, the potential energy surface (PES), which is usually a complicated high-dimensional function.

In principle, this replacement of the electronic structure problem by an analytic PES can be rationalized by quantum mechanics. Since the electronic Hamilton operator is fully defined by the atomic positions, the nuclear charges, and the number of electrons, it is possible to express the PES of a given system using this information only. This, however, is far from trivial, because it is often a tedious task to derive tractable analytic functional forms for high-dimensional PESs without compromising the accuracy. Generations of scientists have developed countless potentials of various forms and complexity for all imaginable systems and purposes. How-

ever, the currently available potentials are all far from satisfying all the needs.

The most important requirements are:^[10]

- a very high accuracy that is comparable to first principles methods, including high-order many-body effects,
- the ability to describe chemical reactions and arbitrary atomic configurations,
- a very high efficiency to enable the simulation of large systems,
- a general applicability to all types of bonding and atomic interactions, from dispersion interactions via covalent bonds to metallic bonding,
- a strategy for systematic improvements, validation, and error control,
- a general automatic construction protocol,
- the absence of ad hoc approximations or system specific energy contributions which restrict the applicability to certain types of systems.

Deriving potentials that fulfill as many of these requirements as possible is currently a very active field of research. Over the years a large variety of approaches to construct reactive PESs has been developed for many different systems,^[11–16] and it is impossible to provide a comprehensive overview in the present Review. The common feature of many of these methods is that the computational costs are reduced by the introduction of physical approximations.

Despite numerous successful applications of these “physical potentials”, in recent years a fundamental change has taken place in the construction of interatomic potentials. In a new generation of potentials, instead of using physical approximations, very flexible functional forms are fitted to high-level electronic structure data.^[17–21] Here, it is not a preselected functional form that is responsible for the correct physical shape of the PES, but a rather large number of parameters that are adjusted by using the physical information contained in first principles reference data. Nowadays, machine learning (ML) methods have taken the lead in this class of “mathematical potentials”.^[22–26] Starting with artificial neural networks (NNs) in 1995,^[27] many different machine learning methods such as Gaussian processes,^[28–30] kernel ridge regression,^[31] and support vector machines^[32,33] have been proposed for the construction of potentials, and many new ideas are continuously emerging,^[34–38] including the vision of force fields augmented by machine learning.^[39]

In this Review, the challenges and solutions for the development of accurate and efficient ML potentials will be illustrated by using neural network potentials (NNPs), which are currently—in terms of applications—one of the most advanced examples. NNPs have already been applied successfully to a series of systems (see Section 5), which demonstrates their usefulness in answering questions that require first principles accuracy, but cannot be addressed directly by electronic structure calculations. Two comprehensive reviews on NNPs are available that cover, in particular, the early literature.^[10,40] The goal of the present Review is to discuss recent developments in the field of neural network potentials with a focus on high-dimensional neural network



Jörg Behler graduated in chemistry at the Universität Dortmund. In 2004 he obtained his PhD at the Fritz-Haber-Institute in Berlin and was awarded the Otto-Hahn medal of the Max-Planck society for his thesis. After postdoctoral research at the ETH Zürich with Prof. Parrinello, in 2007 he established his own research group at the Ruhr-Universität Bochum. In 2013 he was awarded the Hans G. A. Hellmann prize for theoretical chemistry and completed his Habilitation in theoretical chemistry in 2014. Since 2017 he has been a full professor for theoretical chemistry at the Universität Göttingen.

potentials (HDNNPs), which are used by an increasing number of research groups to study large condensed systems, from bulk solids via surfaces and interfaces to solutions.

2. Machine Learning Potentials

Machine learning (ML) is omnipresent in everyday life. It has evolved tremendously in recent years, driven by the needs for the analysis of huge amounts of data generated in the IT industry, and new applications are constantly being found.

The first definition of ML was given by Arthur Samuel, a pioneer in ML research, in 1959: “[ML is a] Field of study that gives computers the ability to learn without being explicitly programmed”. A modern definition can be found in the textbook by Mitchell:^[41] “A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P , if its performance at tasks in T , as measured by P , improves with the experience E . ”

ML methods have also been used successfully in chemistry and physics for a long time,^[42–44] mainly for data processing and classification, such as the analysis of NMR^[45] and mass spectra,^[46] the modeling of crystallization kinetics,^[47] and protein structure prediction.^[48,49] As an example, Figure 1 shows the number of publications in chemistry, physics, and materials science that have made use of artificial neural networks.

In 1995, Blank et al. proposed to use NNs in the representation of DFT PESs for molecule-surface scattering.^[27] This marks the birth of modern ML potentials (MLPs), which can be defined by three criteria:

1. In MLPs, the analytic structure-energy relationship is expressed using a ML method.
2. MLPs are constructed using a consistent reference set of first principles energies (and forces).
3. MLPs do not contain any physical approximations or ad hoc assumptions apart from the electronic structure method chosen to generate the reference data set.

The ML method employed in the case of NNPs is an artificial NN, which will be introduced in detail in Section 3. The construction of any MLP consists of several steps, which

are summarized in Figure 2. First, an electronic structure method is chosen as a reference, which is able to capture the correct physical behavior of the system. This is very important, because the goal of the subsequent construction of the

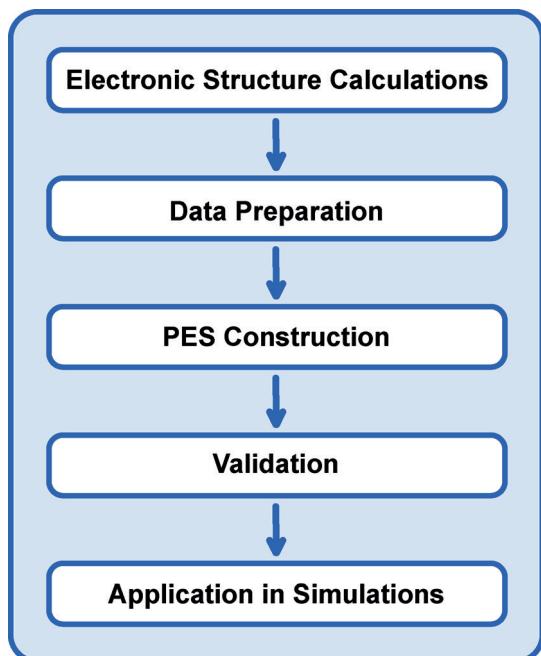


Figure 2. Steps in the construction of machine learning potentials.

MLP is to reach a close numerical agreement of the energies (and forces) with this reference method. If the latter is not sufficiently accurate for the intended application, neither will be the MLP. The electronic structure method is then used to determine the energies, and sometimes the forces, of a rather large set of representative atomic configurations. The construction of this reference set is often the computational bottleneck in the construction of MLPs, and the structures must be carefully chosen to ensure that the important features of the PES are present in the data set. In a second step, the data are prepared for training the ML method. This is done by transforming the atomic positions into a special set of coordinates that has to meet several requirements. This step, which is of utmost importance, will be explained in Section 3.2. It is followed by the fitting process, in which the parameters of the ML method are adjusted to reproduce the reference data as accurately as possible. Before the potential can be applied in simulations, a careful validation is required. This is an essential step, because the functional forms of ML methods have no physical meaning, and the correct shape of the PES has to be learned from the available electronic structure data. Often, it is necessary to include additional structures in the training set and to improve the MLP iteratively. Once it has been carefully checked, it is ready for applications. In recent years, much progress has been made in all these individual tasks.

MLPs offer a number of advantages. They are very accurate and total energy errors as low as a few meV per atom can be achieved. They are several orders of magnitude faster

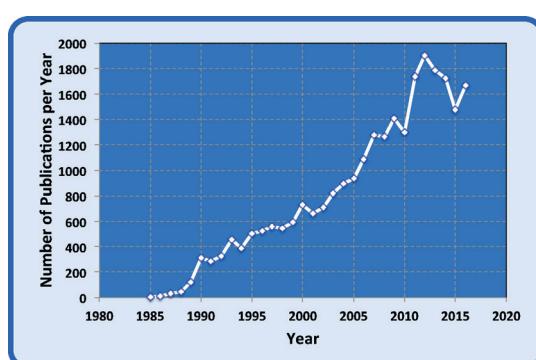


Figure 1. Peer-reviewed publications in chemistry, physics, and materials sciences that made use of artificial neural networks. The data were extracted from the Web of Science^[50] in March 2017.

to evaluate than even efficient electronic structure methods such as DFT, so they can be routinely applied to MD simulation times of several nanoseconds for systems containing tens of thousands of atoms. Another important property which is related to their unbiased mathematical functional form is that MLPs are applicable to all types of systems, such as small molecules,^[51–53] liquids,^[54,55] bulk metals and their surfaces,^[56] semiconductors,^[57,58] and insulators,^[59] irrespective of the physical nature of the bonding and the atomic interactions. As a consequence of their ability to establish a direct relationship between the atomic positions and the energy, MLPs can deal with arbitrary atomic configurations and no specification of bonds or atom types, which is a major limitation of simple force fields, is required. This universal applicability makes MLPs promising candidates for systems which are very challenging for conventional potentials.

On the other hand, the construction of MLPs is very demanding, because large data sets often containing tens of thousands of electronic structure energies are required. Furthermore, since the functional form of ML methods is more complex than that of force fields, they are about one to two orders of magnitude slower. Most importantly, as discussed above, MLPs can fail spectacularly if they have not been properly tested, which is the price to be paid for the high flexibility that enables their accuracy.

3. Neural Network Potentials

Artificial neural networks (NNs) are the oldest ML method.^[60,61] As their name suggests, NNs are biology-inspired functions, and the first artificial neurons were proposed as early as 1943 as mathematical tools to understand signal processing in the human brain.^[62] In analogy to the stimulation of neurons in the nervous system, they were originally designed to provide a threshold-dependent “all-or-nothing” output, which is very useful for data classification—one of the main applications of NNs. In the following years NNs have evolved by the introduction of single-layer perceptrons as a weighted combination of artificial neurons.^[63] These early NNs, however, had some limitations, which turned out to be highly problematic for applications to neural computing in the emerging field of computer science. The most prominent example was given by Minsky and Papert, who showed in 1969 that simple perceptrons are unable to compute some Boolean functions,^[64] which are essential for general computing. As a consequence of this finding, the enthusiasm for NNs decayed substantially and resulted in a decade of very little research activity. Despite these difficulties, several key developments, such as the introduction of hidden layers, of modern training algorithms, and of nonlinear activation functions, contributed to overcoming the limitations, which finally resulted in a renaissance of NN research in the 1980s. In the following years, NNs found countless applications, from weather forecasting to character recognition software, and they also turned out to be useful in chemistry.^[42]

In recent years, the impact of NNs has further increased enormously, as they are now a central tool in the analysis of

“big data” in the form of deep learning, a development which is strongly driven by the needs of the large internet companies.

Over the years, many different types of NNs have been developed and used for different purposes. Apart from their excellent suitability for data classification, it has been proven that NNs are, in principle, able to represent unknown multidimensional real-valued functions with arbitrary precision.^[65,66] This property is one of the two theoretical foundations for their applicability to the construction of PESs. The second is the Born Oppenheimer approximation of quantum mechanics,^[67] which states that the energy of a system can be given as a parametric function of the nuclear positions.

3.1. Early Potentials with Feed-Forward Neural Networks

Essentially all the NN potentials reported since the seminal work by Blank et al. in 1995^[27] contain one or more feed-forward NNs. These are used to define the relationship between the structure of the system and the desired property, which in the case of a PES is the potential energy of the system.

A feed-forward NN, as shown schematically in Figure 3, consists of a number of artificial neurons—or nodes—arranged in layers. The nodes in the input layer provide the structural description in the form of a coordinate vector $\mathbf{G} = (\{G_i\})$ that has to fulfill certain properties, as discussed in Section 3.2. The potential energy E in the node of the output layer is an analytic function of \mathbf{G} . The specific functional form of the NN is defined by the number of hidden layers, which have no physical meaning, and the number of neurons per hidden layer in between the input and the output layer. The more hidden layers and nodes per layer that are present, the

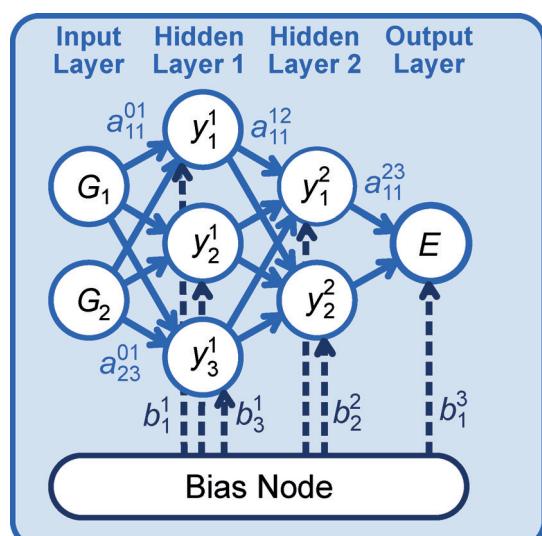


Figure 3. Schematic structure of a feed-forward neural network defining the functional relationship between a two-dimensional input vector $\mathbf{G} = (G_1, G_2)$ that describes the atomic configuration and the potential energy E [Eq. (3)].

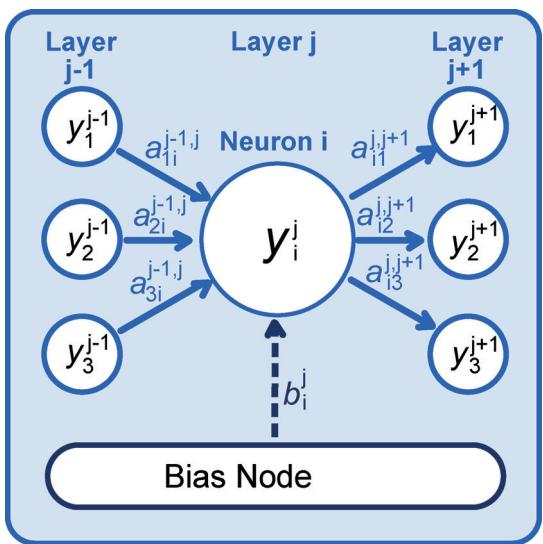


Figure 4. Structure of an artificial neuron i in layer j . First, a weighted sum of the node values y_k^{j-1} in the preceding layer $j-1$ is determined using the connecting weights $\{a\}$ as coefficients. After adding a bias weight b_i^j , an activation function is applied to yield the value y_i^j of the neuron [Eq. (2)]. Finally, y_i^j is forwarded to the next layer of the NN for further processing.

more flexible is the functional form. To determine the value y_i^j of a general node i in layer j (Figure 4), first a linear combination of the N_{j-1} values of the nodes in the preceding layer $j-1$ is calculated [Eq. (1)].

$$x_i^j = b_i^j + \sum_{k=1}^{N_{j-1}} a_{k,i}^{j-1,j} \cdot y_k^{j-1} \quad (1)$$

In the case of the first hidden layer ($j=1$), the preceding layer is the input layer, so that y_k^{j-1} corresponds to G_k . The relative weight of each supplied value is controlled by the NN weight parameter $a_{k,i}^{j-1,j}$ that connects node k in layer $j-1$ to node i in layer j . The bias weight b_i^j that is added to this linear combination has the purpose of an adjustable offset. Finally, an activation function f_i^j is applied to the result of Equation (1) to yield Equation (2).

$$y_i^j = f_i^j(x_i^j) = f_i^j\left(b_i^j + \sum_{k=1}^{N_{j-1}} a_{k,i}^{j-1,j} \cdot y_k^{j-1}\right) \quad (2)$$

Activation functions are continuous nonlinear functions that substitute the step functions in the early NNs that had been used to mimic the activation thresholds of biological neurons. The use of continuous functions enables the binary output of NNs to be replaced by continuous values, as are needed for functions such PESs. Many types of activation functions can be used, such as the hyperbolic tangent, the sigmoid function, or Gaussian functions. These functions have the property of converging to constant numbers for very positive and very negative arguments, while in between they possess a nonlinear region, which enables the NN to represent functions of arbitrary shape. Only in the case of the output

node the linear function $y=x$ is often used to avoid any restriction in the range of possible output energy values.

The full analytical form of the feed-forward NN shown in Figure 3 containing $N_{\text{hid},1}=3$ and $N_{\text{hid},2}=2$ nodes in the two hidden layers, as well as $N_G=2$ input nodes is given by Equation (3).

$$E = f_1^3 \left(b_1^3 + \sum_{k=1}^{N_{\text{hid},2}} a_{k1}^{23} \cdot f_2^k \left(b_k^2 + \sum_{l=1}^{N_{\text{hid},1}} a_{lk}^{12} \cdot f_1^l \left(b_l^1 + \sum_{i=1}^{N_G} a_{ii}^{01} \cdot G_i \right) \right) \right) \quad (3)$$

The input layer has the superscript 0. Overall, the NN has a nested hierarchical functional form of rather simple activation functions that act on linear combinations. A demonstration of the resulting fitting capabilities can be found in Ref. [68] for simple model functions.

Apart from the atomic positions specified by $\{G_i\}$, the energy of the system depends on the numerical values of the connecting weight parameters $\{a\}$ and the bias weights $\{b\}$. These parameters are determined in a gradient-based iterative optimization by using a set of known reference energies—and sometimes forces—from electronic structure calculations. In principle, a wide range of gradient-based optimization algorithms can be used, but in the case of NNs the “back propagation” algorithm^[69] and the global adaptive Kalman filter,^[70,71] in particular, are very popular. The overall goal of this optimization is to minimize the root mean squared error (RMSE) of the energies (and of the forces) of a given training set. Since the number of weight parameters N_w is often in the order of a few thousand, it is impossible to find the global minimum in this optimization process. However, it is not difficult to find numerous acceptable local minima. Training NNPs is very demanding and can take several weeks, particularly when using large data sets.

The flexibility of the NN is of paramount importance for the ability to represent the target PES. It depends on the number of weight parameters, which is given by the architecture of the NN, namely, by the number of hidden layers and neurons per layer. If the NN flexibility is too low (see Figure 5 a), the fitting error is large and the NN is not able to reproduce the PES accurately. If more neurons are used, a good representation of the PES is obtained (Figure 5 b). If even more parameters are included by increasing the size of the NN, overfitting can occur, which can give rise to a poor overall representation of the PES but cannot be detected by the error of the training points alone (Figure 5 c). There are several options to detect and avoid such overfitting, as discussed in Section 4. In practical applications, it has turned out to be most efficient to construct NNPs with several different NN architectures, to analyze the results, and to select the best potential for the application. Typical NNPs contain one to three hidden layers and up to 50 neurons per layer.

Feed-forward NNs (FFNNs), as described in this section, have been used for almost two decades for many systems, and a list of the studied systems can be found in two earlier reviews.^[10,40] These NNPs suffered from the restriction to low-dimensional systems. This limitation had several reasons:

- The number of input nodes is related to the number of degrees of freedom in the system, but cannot be increased

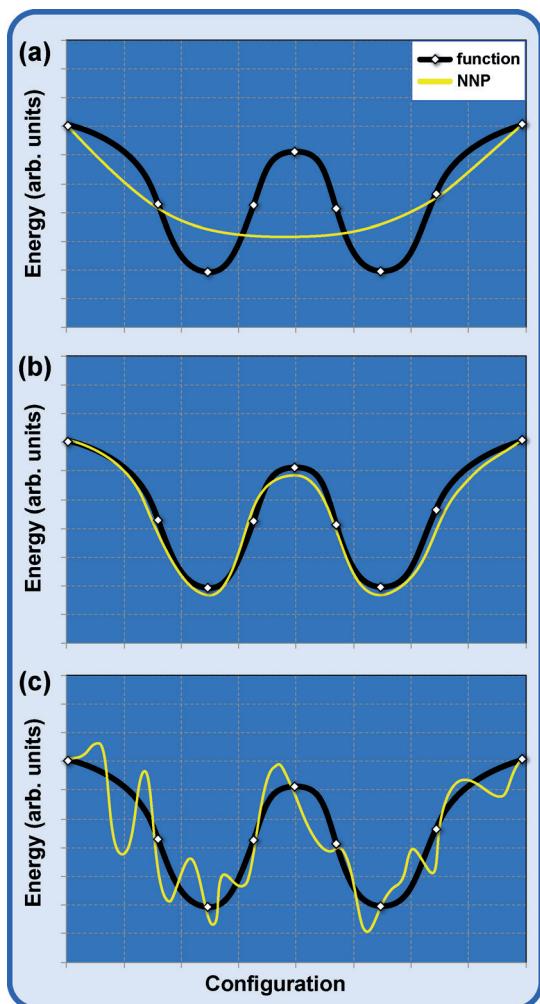


Figure 5. Schematic illustration of the relationship between a) “underfitting”, b) a reasonable fit, and c) overfitting, and the flexibility of the NN. In (a), the NN is not flexible enough to reproduce all the relevant details of the target PES (black line), thereby resulting in a large error. The increased flexibility in (b) allows for a good representation of the PES. In (c), all known example points, shown as white symbols, are still represented accurately, as in (b), but the very high flexibility gives rise to artificial features between these points that cannot be detected by monitoring the error of the training set only.

arbitrarily to prevent the size of the NN from becoming intractable.

- The development of suitable input coordinates for the NNP has been a major challenge, because the energy of the system must be rotationally and translationally invariant. Furthermore, permutation symmetry with respect to the exchange of like atoms must be ensured. All these requirements are not fulfilled by ordered vectors of Cartesian coordinates, and using internal coordinates such as distances and angles instead would only consider the first two invariances. This problem was recognized early on and some solutions for small molecular systems^[72] and molecule-surface scattering^[73,74] have been proposed.
- Since the number of input nodes of a trained NN cannot be changed, an NNP can only be used for one fixed system size.

All these points prevented the application of early NNPs to high-dimensional systems containing more than a few mobile atoms. Nevertheless, a number of promising attempts to extend the applicability of NNPs have been reported, such as the construction of NNPs based on chains of atoms by employing NNs of variable size,^[57,75] the use of multiple NNs and redundant coordinates by employing a high-dimensional model representation,^[51,76] and the combination of NNs with a many-body expansion approach.^[77] A detailed discussion of these methods can be found in Ref. [10].

3.2. High-Dimensional Neural Network Potentials

An NNP method, which is applicable to high-dimensional systems containing thousands of atoms was proposed by Behler and Parrinello in 2007.^[78] In this approach,^[79,80] the conceptual problems of early NNPs discussed in the previous section were overcome by using a separate FFNN for each atom in the system. Each of these “atomic NNs” provides the energy contribution E_μ of atom μ as a function of the chemical environment. The total (short range) energy E_S of the system is then obtained as the sum over all N_{atoms} atoms in the system [Eq. (4)].

$$E_S = \sum_{\mu=1}^{N_{\text{atoms}}} E_\mu \quad (4)$$

For a multicomponent system containing N_{elem} chemical elements, this equation becomes a sum of all the atomic energy contributions of all the elements [Eq. (5)].

$$E_S = \sum_{\nu=1}^{N_{\text{elem}}} \sum_{\mu=1}^{N_{\text{atoms},\nu}} E_\mu^\nu \quad (5)$$

For a given element, the atomic NNs are constrained to have the same architecture—specifying the number of hidden layers and neurons—and the same weight parameters. The structure of the resulting HDNNP is shown in Figure 6 for a ternary system containing elements A, B, and C.

The input of each atomic NN is a vector of atom-centered symmetry functions^[81] describing the local chemical environments of the atoms. These are defined by a cutoff radius R_c , as illustrated in Figure 7. R_c , which has values between 6 and 10 Å, is a convergence parameter that needs to be increased until all the energetically relevant interactions are included, since atoms outside R_c do not enter the energy contribution of the respective central atom.

Several types of symmetry functions are available.^[81] These are all many-body functions that depend simultaneously on the positions of all the atoms inside R_c . Their numerical values are invariant with respect to rotation and translation as well as to the order of the neighboring atoms and, thus, possess all the required invariances. This is a necessary condition to ensure that all geometrically and, thus, energetically equivalent representations of a system are described by the same numerical values of an NN input vector of predetermined symmetry functions. Furthermore, they

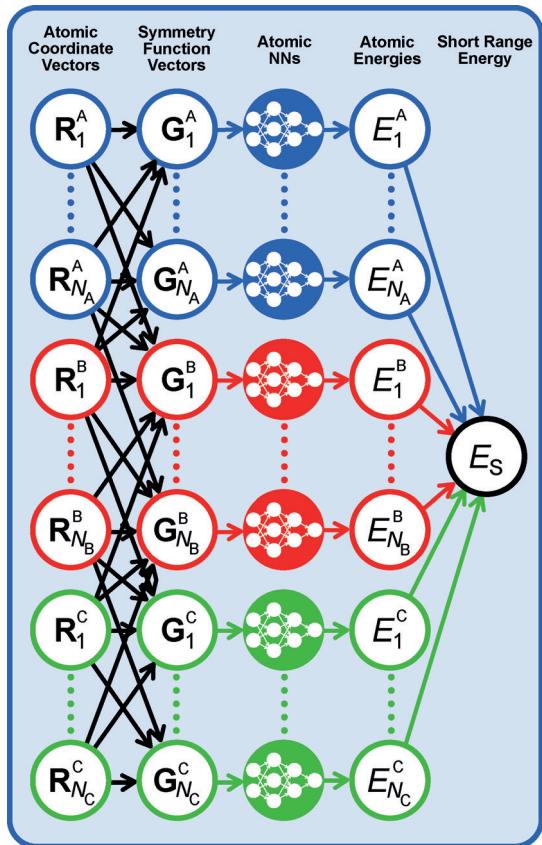


Figure 6. High-dimensional neural network for a ternary system containing elements A, B, and C.^[78,80] The numbers of atoms per element are N_A , N_B , and N_C , respectively. The total short range energy E_S is the sum of all atomic energies E_i^X ($X=A,B,C$), which are provided by individual atomic NNs [Eq. (5)]. For a given element, the architecture and parameters of the atomic NNs are the same. The symmetry function vectors G_i^X provide the information about the local chemical environments of the atoms to the atomic NNs. Consequently, G_i^X depends on the Cartesian position vectors R_i^X of all the atoms within the cutoff spheres, which is represented by the black arrows. For clarity, only some of these arrows are shown.

must be continuous and differentiable to enable the calculation of analytic derivatives for the forces. At the cutoff radius, they decay to zero in both the value and slope according to the cutoff function [Eq. (6)], where R_{ij} is the distance between the central atom i and its neighbor j .

$$f_c(R_{ij}) = \begin{cases} 0.5 \cdot \left[\cos\left(\frac{\pi R_{ij}}{R_c}\right) + 1 \right] & \text{for } R_{ij} \leq R_c \\ 0 & \text{for } R_{ij} > R_c \end{cases} \quad (6)$$

As a consequence of the requirement that the atomic NNs have a fixed number of input nodes, the number of symmetry functions in the G_i vectors must not change with the number of atoms in the cutoff sphere, even if this number increases or decreases, for example, in molecular dynamics simulations. This can be achieved by using the “radial” symmetry function defined by Equation (7).

$$G_i^2 = \sum_j e^{-\eta(R_{ij}-R_s)^2} \cdot f_c(R_{ij}) \quad (7)$$

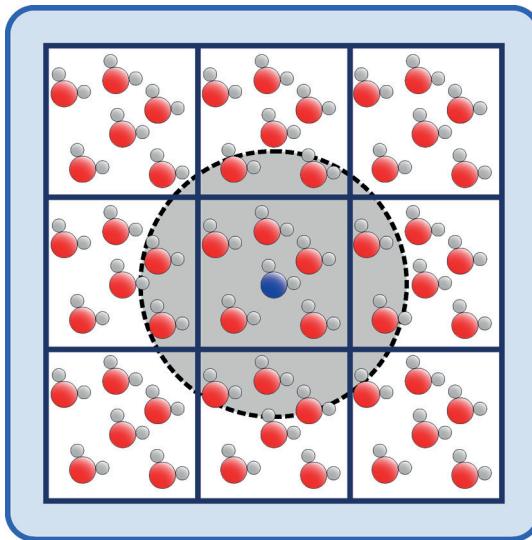


Figure 7. Illustration of the environment-dependence of the atomic energies in Equation (5) for a system with periodic boundary conditions. The energy contribution of the blue atom depends on the positions of all the atoms within the dashed sphere that is defined by the cutoff radius R_c of the symmetry functions.

This is a sum of products of a Gaussian function of the interatomic distance and the cutoff function, which allows for a physical interpretation as the effective coordination number of the central atom. The typical use of 5–6 radial functions with different Gaussian exponents η provides a radial fingerprint of the neighboring atoms. The parameter R_s can be used to shift the centers of the Gaussians to specific interatomic distances. In the case of multicomponent systems, one set of radial functions is used for every neighboring element in the system. Since the radial functions alone are unable to distinguish different angular arrangements of neighbors, a set of “angular functions” [Eq. (8)] should be used, which

$$G_i^4 = 2^{1-\zeta} \sum_j \sum_{k \neq j} (1 + \lambda \cdot \cos\theta_{ijk})^\zeta \cdot f_c(R_{ij}) f_c(R_{ik}) f_c(R_{jk}) \quad (8)$$

depends on the angles θ_{ijk} centered at atom i and formed with neighbors j and k , which both need to be within R_c . The use of a set of functions with different exponents ζ allows a fingerprint of the angular distribution to be obtained, while $\lambda = \pm 1$ can be used to adjust the positions of the maxima and minima of these functions. The properties of these two example functions, for which the superscripts 2 and 4 have been used for consistency with previous work,^[81] and a variety of other symmetry functions are discussed in detail in Ref. [81]. Depending on the chemical complexity of the system, a set of 20 to 100 symmetry functions is typically used to describe the atomic environments.

Since the NNP has a well-defined functional form, the forces can be calculated analytically as the negative gradients of the energy. The force acting on component $\alpha = (X, Y, Z)$ of atom i is given by Equation (9), where two indices μ and ν of

$$F_{\alpha_i} = -\frac{\partial E_S}{\partial \alpha_i} = -\sum_{\mu=1}^{N_{\text{atoms}}} \frac{\partial E_{\mu}}{\partial \alpha_i} = -\sum_{\mu=1}^{N_{\text{atoms}}} \sum_{\nu=1}^{N_{G,\mu}} \frac{\partial E_{\mu}}{\partial G_{\mu\nu}} \cdot \frac{\partial G_{\mu\nu}}{\partial \alpha_i} \quad (9)$$

the symmetry functions have now been used to label the atom and the number of the symmetry function, respectively. The derivatives of the atomic energies with respect to the symmetry functions can be obtained from the architectures of the atomic NNs, while the derivatives of the symmetry functions with respect to α_i are given by the symmetry function definitions. It is interesting to note that as a consequence of Equation (9), the dependence of the forces on the atomic environment has a larger effective range than the atomic energies (Figure 8). This is a necessary condition to achieve consistent energies and forces, which is important for energy-conserving molecular dynamics simulations.

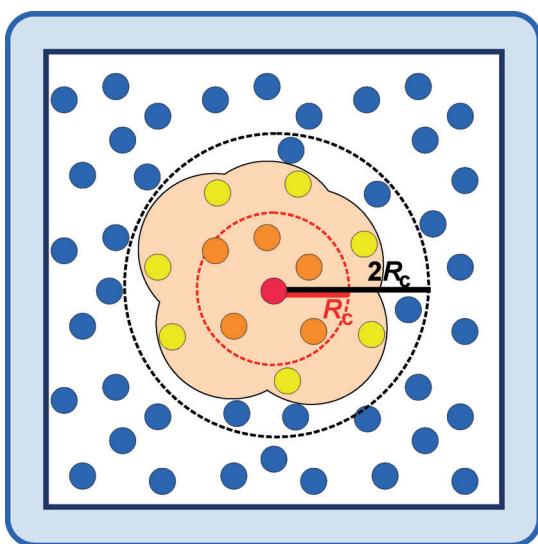


Figure 8. Schematic illustration of the environment-dependence of the force vector acting on the red atom in the center. Since the red atom is within the symmetry function cutoff radius R_c of all the orange atoms, the force depends also on the positions of the yellow atoms, which are in the chemical environments of the orange atoms, but beyond R_c . Therefore, the positions of the red and the yellow atoms are coupled, as they both enter the atomic energy contributions of the orange atoms [Eq. (5)]. This effectively extends the environment dependence of the forces to a maximum radius of $2R_c$ depending on the specific distance of the orange atoms from the central atom. Please note that while this is a formal consideration that ensures consistency between the forces and the total energy, the numerical influence of the atoms beyond R_c may be very small for large symmetry function cutoffs.

The training of the NN parameters can be carried out using total energies and, if desired, using forces, which provide valuable local information about the shape of the PES. It is not required to use individual atomic energies obtained from energy partitioning schemes.

For some systems, such as ionic materials with substantial long-range electrostatic interactions, it might be desirable to avoid a truncation of the atomic interactions at R_c . This is possible by adding electrostatic interactions explicitly using a second set of atomic NNs, in the same way as in Figure 6, to determine environment-dependent atomic charges,^[82,83] so that the total energy is described by Equation (10).

$$E_{\text{tot}} = E_S + E_{\text{elec}} \quad (10)$$

An alternative way of deriving general electrostatic multipoles by using machine learning has been proposed by Popelier and co-workers.^[84,85] A very promising approach for HDNNPs based on charge equilibration has recently also been suggested, which does not require the use of reference charges from electronic structure calculations.^[86,87]

HDNNPs of this form offer several advantages compared to conventional NNPs. They satisfy all the required invariances exactly, and the atom-centered symmetry functions are applicable to all types of systems. Even the choice of the specific parameters defining the shape of the symmetry functions is surprisingly insensitive to the specific systems. As a consequence of the locality of the atomic interactions, it is possible to train HDNNPs by using small systems containing only a few hundred atoms, and once constructed, they can be applied to much larger systems, as these are effectively decomposed into local motifs by the NNP.

4. Discussion

The structure of MLPs is very modular and essentially they consist of two main components, a structural descriptor and an ML method. HDNNPs make use of many-body atom-centered symmetry functions^[81] to describe the atomic environments and a set of atomic NNs to connect the descriptor vectors to energies. There are many other possible choices for both components. Other descriptors include four-dimensional spherical harmonics in combination with Gaussian processes,^[28,88] Coulomb matrices for use with kernel ridge regression,^[31,89] and molecular local frames and NNs for electrostatic multipoles.^[85] New ideas for descriptors are constantly published, such as the smooth overlap of atomic positions (SOAP) method,^[90] the bag of bonds approach,^[91] symmetry functions augmented by low-dimensional functions,^[92] and Fourier series of atomic radial distribution functions.^[93] Furthermore, there is also a close relationship to the challenges in structure identification, and while atom-centered symmetry functions developed for NNPs have been used for the detection of local structure^[94] and for the characterization of disordered materials,^[95,96] MLPs may, in turn, also benefit from developments in other fields, which is illustrated, for example, by the combination of permutation-invariant polynomials and NNs.^[97,98] The availability of all these methods has substantially contributed to the success of MLPs, since the identification of suitable descriptors has been one of the main problems faced by the developers of early MLP techniques.

A remaining challenge of today's MLPs is their limited transferability to new atomic configurations. This problem is illustrated in Figure 9 for a one-dimensional model system. While the potential shown as a white line is close to the reference energies in the interval $[G_{\min}, G_{\max}]$ defined by the training points, the uncertainty of the predicted energy increases rapidly outside of this interval. This case of extrapolation is easily detected, as this interval is known for every descriptor from the training process. Unfortunately, there are more complex situations in which there are less-well-represented regions within these intervals, as shown in

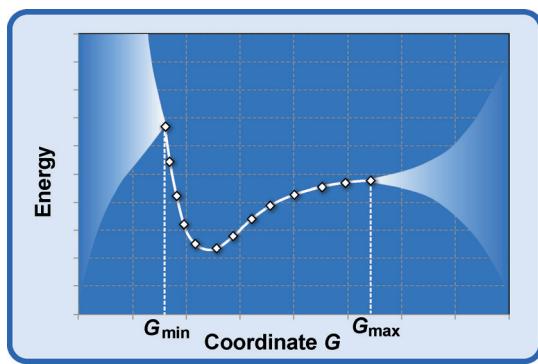


Figure 9. Similar to any ML method, NNs have limited extrapolation capabilities and predictions for structures beyond the known reference configurations represented by the white symbols, that is, outside the interval $[G_{\min}, G_{\max}]$, can be unreliable.

Figure 10. Such situations may easily remain undetected, thereby giving rise to large errors in the predicted energies and forces. A possible solution to identify these insufficiently sampled regions and to add missing structures to the training set can be derived by exploiting the high flexibility of MLPs, which has been demonstrated extensively for NNPs.^[56,83] In this approach, several NNPs with different functional forms are constructed for a given reference set. Then, two or more NNPs with comparable RMSEs for the training structures are selected. One of them is used to generate a large number of configurations for screening, for example, by MD or Monte Carlo simulations. Then, the energies and forces of these configurations are compared to the corresponding values obtained using the other potentials. For structures which are close to the underlying training data, all the potentials are

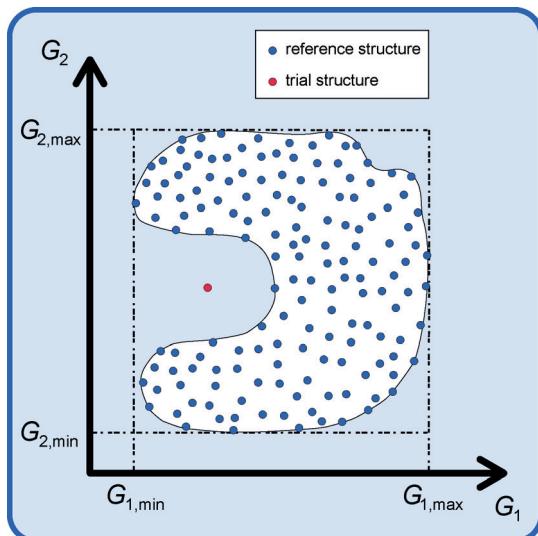


Figure 10. Illustration of an insufficiently sampled region in a two-dimensional symmetry function space. The trial structure (red symbol) is within the intervals $[G_{1,\min}, G_{1,\max}]$ and $[G_{2,\min}, G_{2,\max}]$ but outside the white region, which is covered well by the blue reference structures. Therefore, the predicted energy may have a high uncertainty and additional structures close to the trial structure should be added to the training set following the procedure described in Figure 11.

likely to predict very similar energies and forces. In insufficiently sampled regions, the energies and forces will differ substantially. Such structures should then be included in the training set. In this way, the training set can be iteratively improved until converged potentials are obtained. This procedure is illustrated in Figure 11.

In summary, the validation of MLPs consists of several steps:

1. It must be checked whether the structure of interest is within the descriptor range spanned by the training set for each coordinate. If extrapolation is detected, the validity range of the PES needs to be extended.
2. In the fitting process, the early stopping method should be used to avoid overfitting. In this approach, not all the available reference structures are used, but the data are split into a training set used to optimize the parameters of the MLP, while an independent test set is used to check the

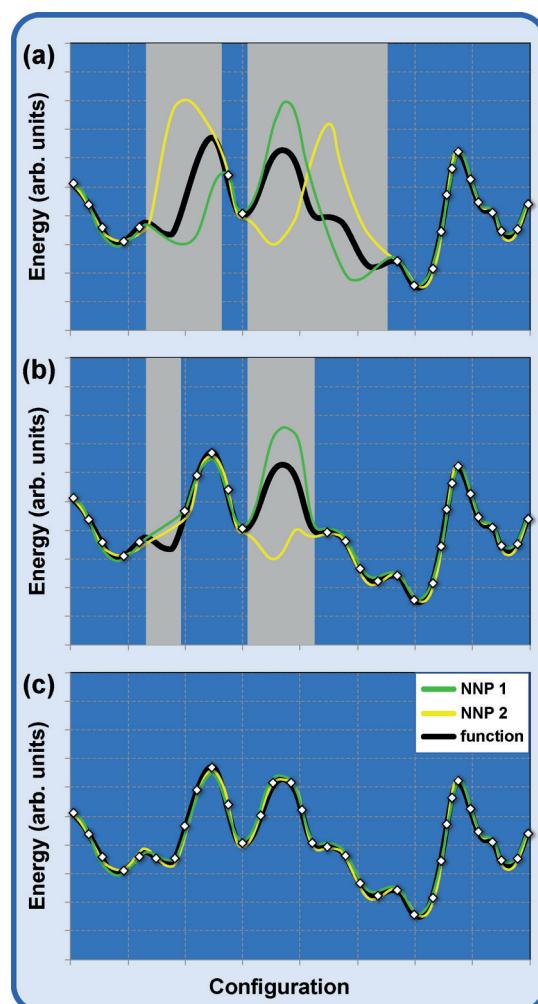


Figure 11. Iterative improvement of the training set using two NNPs. In (a), only a few structures are included in the training set (white symbols), and wide regions (gray) are not included. Therefore, both NNPs predict very different energies. This difference can be used to identify new structures that should be included in the training set, thereby resulting in a reduction in the gray regions in (b). In (c), both NNPs agree very closely for all configurations, thus indicating that all relevant configurations are well represented in the training set.

accuracy for structures not included in the training. During the fit, the errors of both the training and the test set are monitored, and the fit with the best generalization properties is the one with the lowest error in the test set. More sophisticated cross-validation procedures can also be used in this way.

- The early stopping method is unable to identify insufficiently sampled regions (see Figure 10), because both the training and the test set are derived from the available reference data. Therefore, the predictions of several fits should be compared, and only in the case of good agreement the potential can be trusted.

Despite this hierarchy of validation steps, the efficient sampling of configurational spaces still remains an important research topic.^[99,100]

5. Summary and Outlook

A lot of progress has been made in recent years in the construction of MLPs. Since the introduction of the first method suitable for high-dimensional systems,^[78] other ML methods with similar capabilities have also emerged, most notably—but not only—Gaussian approximation potentials (GAPs).^[28] Other methods are close to reaching this level of applicability. These MLPs close a gap in the toolbox of theoretical chemistry and materials science, in that they can provide very accurate potentials for reactive systems of complex materials. They are complementary to established methods in other fields, such as QM/MM methods, which are the method of choice for local chemical reactions. A particular strength of ML methods is the ability to describe all parts of the systems in the same way so that no previous knowledge on the spatial location of the reaction is needed, which is impossible for many systems. This is particularly useful for simulations in materials science, for processes at interfaces, and for reactions in solutions, in particular if proton transfer plays a role.^[101,102] A list of HDNNPs which have become dominant in the field of NNPs is shown in Table 1. All these examples show that HDNNPs are suitable for systems as diverse as small molecules,^[53] molecular clusters,^[103] metal clusters,^[104,105] bulk materials,^[58] surfaces,^[56] water,^[55,106] aqueous electrolyte solutions,^[107] and solid–liquid interfaces,^[108] and they have contributed to new physical insights.

However, there are also some important disadvantages of MLPs, such as the substantial effort in their development, the need for a careful validation, the requirement of large training sets often containing tens of thousands of structures, the limited transferability to configurations that are very different from the training data, and the current restriction to systems containing about four elements because of the exponentially growing complexity of configuration space with a growing number of elements.

Beyond these applications, many new ideas related to HDNNPs have been proposed, such as the improvement of semiempirical methods for the description of the QM part in QM/MM simulations,^[109] the acceleration of saddle point

Table 1: Applications of high-dimensional neural network potentials.

Year ^[a]	System	References
2007	bulk silicon	[58, 78, 110]
2010	bulk carbon	[59, 111]
2010	bulk sodium	[112, 113]
2011	ZnO	[82]
2012	bulk GeTe	[114–117]
2012	copper	[56–118]
2012	methanol molecule	[118]
2012	water clusters	[83, 103, 119, 120]
2013	Cu clusters on ZnO	[121]
2014	Au/Cu clusters and water	[122]
2015	Cu-Au nanoalloys	[123]
2015	allyl vinyl ether	[53]
2016	bulk water	[55, 106, 124]
2016	Cu on CeO ₂	[125]
2016	water on copper	[108, 126]
2016	n-alkanes	[127]
2016	ethane molecule	[128]
2016	Na clusters	[104]
2016	H ₂ +SH	[129]
2016	H ₂ +H, H ₂ O+H, CH ₄ +H	[130]
2016	TiO ₂	[131]
2016	bulk gold and surfaces	[132]
2016	NaOH in water	[101, 107]
2017	Cu/Pd/Ag	[133]
2017	HCl at Au(111)	[134]
2017	AuPd(111) surface	[135]
2017	CaF ₂	[87]
2017	oxygen on Pd surface	[136]
2017	Au clusters	[105]
2017	water on ZnO	[102]

[a] In the case of several publications for a particular system, the year of the first publication is given.

searches,^[127] a hierarchical training of NNs for multicomponent systems,^[133] the prediction of NMR parameters of solid-state materials,^[137] the prediction of heats of formation,^[138] and a new approach through a combination with many-body expansions.^[139] Many of these developments have been published only very recently, thus showing that the field is rapidly evolving and many new interesting developments can be expected to further extend the range of applications of NNPs and MLPs in the near future.

Acknowledgements

The author thanks the DFG for a Heisenberg Professorship (Be3264/11-1).

Conflict of interest

The author declares no conflict of interest.

How to cite: *Angew. Chem. Int. Ed.* **2017**, *56*, 12828–12840
Angew. Chem. **2017**, *129*, 13006–13020

- [1] A. Ruzsinszky, J. P. Perdew, *Comput. Theor. Chem.* **2011**, *963*, 2–6.
- [2] R. G. Parr, W. Yang, *Density Functional Theory of Atoms and Molecules*, Oxford University Press, Oxford, **1989**.
- [3] R. Car, M. Parrinello, *Phys. Rev. Lett.* **1985**, *55*, 2471–2474.
- [4] D. Marx, J. Hutter *Ab initio Molecular Dynamics: Basic Theory and Advanced Methods*, Cambridge University Press, Cambridge, **2009**.
- [5] A. Szabo, N. S. Ostlund, *Modern Quantum Chemistry—Introduction to Advanced Electronic Structure Theory*, Dover, New York, **1996**.
- [6] A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard III, W. M. Skiff, *J. Am. Chem. Soc.* **1992**, *114*, 10024–10035.
- [7] N. L. Allinger, Y. H. Yuh, J.-H. Lii, *J. Am. Chem. Soc.* **1989**, *111*, 8551–8566.
- [8] B. R. Brooks, R. E. Bruckolieri, B. D. Olafson, D. J. States, S. Swaminathan, M. Karplus, *J. Comput. Chem.* **1983**, *4*, 187–217.
- [9] H. M. Senn, W. Thiel, *Top. Curr. Chem.* **2007**, *268*, 173–290.
- [10] J. Behler, *Phys. Chem. Chem. Phys.* **2011**, *13*, 17930–17955.
- [11] J. Tersoff, *Phys. Rev. Lett.* **1986**, *56*, 632–635.
- [12] J. F. Justo, M. Z. Bazant, E. Kaxiras, V. V. Bulatov, S. Yip, *Phys. Rev. B* **1998**, *58*, 2539–2550.
- [13] T. Hammerschmidt, R. Drautz, D. G. Pettifor, *Int. J. Mater. Res.* **2009**, *100*, 1479–1487.
- [14] M. Elstner, D. Porezag, G. Jungnickel, J. Elsner, M. Haugk, T. Frauenheim, S. Suhai, G. Seifert, *Phys. Rev. B* **1998**, *58*, 7260.
- [15] A. C. T. van Duin, S. Dasgupta, F. Lorant, W. A. Goddard III, *J. Phys. Chem. A* **2001**, *105*, 9396–9409.
- [16] V. Babin, C. Leforestier, F. Paesani, *J. Chem. Theory Comput.* **2013**, *9*, 5395–5403.
- [17] M. J. T. Jordan, K. C. Thompson, M. A. Collins, *J. Chem. Phys.* **1995**, *103*, 9669.
- [18] D. E. Makarov, H. Metiu, *J. Chem. Phys.* **1998**, *108*, 590–598.
- [19] G. L. W. Hart, V. Blum, M. J. Walorski, A. Zunger, *Nat. Mater.* **2005**, *4*, 391.
- [20] B. J. Braams, J. M. Bowman, *Int. Rev. Phys. Chem.* **2009**, *28*, 577–606.
- [21] T. Ishida, G. C. Schatz, *Chem. Phys. Lett.* **1999**, *314*, 369–375.
- [22] C. M. Handley, J. Behler, *Eur. Phys. J. B* **2014**, *87*, 152.
- [23] J. Behler, *J. Chem. Phys.* **2016**, *145*, 170901.
- [24] L. Ward, C. Wolverton, *Curr. Opin. Solid State Mater. Sci.* **2017**, *21*, 167–176.
- [25] V. Botu, R. Batra, J. Chapman, R. Ramprasad, *J. Phys. Chem. C* **2017**, *121*, 511–522.
- [26] T. Mueller, A. G. Kusne, R. Ramprasad, *Rev. Comput. Chem.* **2016**, *29*, 186.
- [27] T. B. Blank, S. D. Brown, A. W. Calhoun, D. J. Doren, *J. Chem. Phys.* **1995**, *103*, 4129–4137.
- [28] A. P. Bartók, M. C. Payne, R. Kondor, G. Csányi, *Phys. Rev. Lett.* **2010**, *104*, 136403.
- [29] A. P. Bartók, G. Csányi, *Int. J. Quantum Chem.* **2015**, *115*, 1051–1057.
- [30] M. J. L. Mills, P. L. A. Popelier, *Theor. Chim. Acta* **2012**, *131*, 1137.
- [31] M. Rupp, A. Tkatchenko, K.-R. Müller, O. A. von Lilienfeld, *Phys. Rev. Lett.* **2012**, *108*, 058301.
- [32] A. Vitek, M. Stachon, P. Krömer, V. Snasel, IEEE 5th International Conference on Intelligent Networking and Collaborative Systems **2013**, 121.
- [33] R. M. Balabin, E. I. Lomakina, *Phys. Chem. Chem. Phys.* **2011**, *13*, 11710.
- [34] A. V. Shapeev, *Multiscale Model. Simul.* **2016**, *14*, 1153–1173.
- [35] M. Hirn, S. Mallat, N. Poilvert, *Multiscale Model. Simul.* **2017**, *15*, 827.
- [36] R. Ramakrishnan, P. O. Dral, M. Rupp, O. A. von Lilienfeld, *J. Chem. Theory Comput.* **2015**, *11*, 2087–2096.
- [37] Z. Li, J. R. Kermode, A. De Vita, *Phys. Rev. Lett.* **2015**, *114*, 096405.
- [38] E. V. Podryabinkin, A. V. Shapeev, arXiv:1611.09346.
- [39] P. L. A. Popelier, *Int. J. Quantum Chem.* **2015**, *115*, 1005–1011.
- [40] C. M. Handley, P. L. A. Popelier, *J. Phys. Chem. A* **2010**, *114*, 3371–3383.
- [41] T. M. Mitchell, *Machine Learning*, McGraw Hill, New York, **1997**.
- [42] J. Gasteiger, J. Zupan, *Angew. Chem. Int. Ed. Engl.* **1993**, *32*, 503–527; *Angew. Chem.* **1993**, *105*, 510–536.
- [43] A. Jain, G. Hautier, S. P. Ong, K. Persson, *J. Mater. Res.* **2016**, *31*, 977–994.
- [44] G. Reibnegger, G. Weiss, G. Werner-Felmayer, G. Judmaier, H. Wachter, *Proc. Natl. Acad. Sci. USA* **1991**, *88*, 11426.
- [45] J. U. Thomsen, B. Meyer, *J. Magn. Reson.* **1989**, *84*, 212–217.
- [46] B. Curry, D. E. Rumelhart, *Tetrahedron Comput. Methodol.* **1990**, *3*, 213–237.
- [47] S. Y. Wong, R. K. Bund, R. K. Connelly, R. W. Hartel, *J. Cryst. Growth Des.* **2010**, *10*, 2620.
- [48] L. H. Holley, M. Karplus, *Proc. Natl. Acad. Sci. USA* **1989**, *86*, 152.
- [49] A. A. Rabow, H. A. Scheraga, *J. Mol. Biol.* **1993**, *232*, 1157.
- [50] Extracted using the Web of Science http://www.webofknowledge.com, Clarivate Analytics, in March 2017.
- [51] S. Manzhos, T. Carrington, Jr., *J. Chem. Phys.* **2008**, *129*, 224104.
- [52] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, K.-R. Müller, *J. Chem. Theory Comput.* **2013**, *9*, 3404–3419.
- [53] M. Gasteiger, P. Marquetand, *J. Chem. Theory Comput.* **2015**, *11*, 2187–2198.
- [54] A. P. Bartók, M. J. Gillan, F. R. Manby, G. Csányi, *Phys. Rev. B* **2013**, *88*, 054104.
- [55] T. Morawietz, A. Singraber, C. Dellago, J. Behler, *Proc. Natl. Acad. Sci. USA* **2016**, *113*, 8368.
- [56] N. Artrith, J. Behler, *Phys. Rev. B* **2012**, *85*, 045439.
- [57] E. Sanville, A. Bholoa, R. Smith, S. D. Kenny, *J. Phys. Condens. Matter* **2008**, *20*, 285219.
- [58] J. Behler, R. Martoňák, D. Donadio, M. Parrinello, *Phys. Rev. Lett.* **2008**, *100*, 185501.
- [59] R. Z. Khalilullin, H. Eshet, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **2010**, *81*, 100103.
- [60] C. M. Bishop *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, **1996**.
- [61] S. Haykin, *Neural Networks and Learning Machines: A Comprehensive Foundation*, 3rd ed., Prentice Hall, Upper Saddle River, **2008**.
- [62] W. McCulloch, W. Pitts, *Bull. Math. Biophys.* **1943**, *5*, 115–133.
- [63] F. Rosenblatt, *Psych. Rev.* **1958**, *65*, 386–408.
- [64] M. Minsky, S. A. Papert *Perceptrons*, MIT, Cambridge, **1969**.
- [65] G. Cybenko, *Math. Control Sign. Syst.* **1989**, *2*, 303–314.
- [66] K. Hornik, M. Stinchcombe, H. White, *Neural Networks* **1989**, *2*, 359–366.
- [67] M. Born, R. Oppenheimer, *Ann. Phys.* **1927**, *389*, 457–484.
- [68] J. B. Witkoskie, D. J. Doren, *J. Chem. Theory Comput.* **2005**, *1*, 14–23.
- [69] D. E. Rumelhart, G. E. Hinton, R. J. Williams, *Nature* **1986**, *323*, 533–536.
- [70] R. E. Kalman, *J. Basic Eng.* **1960**, *82*, 35–45.
- [71] T. B. Blank, S. D. Brown, *J. Chemom.* **1994**, *8*, 391–407.
- [72] H. Gassner, M. Probst, A. Lauenstein, K. Hermansson, *J. Phys. Chem. A* **1998**, *102*, 4596–4605.
- [73] S. Lorenz, A. Groß, M. Scheffler, *Chem. Phys. Lett.* **2004**, *395*, 210–215.
- [74] J. Behler, S. Lorenz, K. Reuter, *J. Chem. Phys.* **2007**, *127*, 014705.

- [75] S. Hobday, R. Smith, J. Belbruno, *Modell. Simul. Mater. Sci. Eng.* **1999**, 7, 397–412.
- [76] S. Manzhos, T. Carrington, Jr., *J. Chem. Phys.* **2006**, 125, 194105.
- [77] M. Malshe, R. Narulkar, L. M. Raff, M. Hagan, S. Bukkapatnam, P. M. Agrawal, R. Komanduri, *J. Chem. Phys.* **2009**, 130, 184102.
- [78] J. Behler, M. Parrinello, *Phys. Rev. Lett.* **2007**, 98, 146401.
- [79] J. Behler, *Int. J. Quantum Chem.* **2015**, 115, 1032–1050.
- [80] J. Behler, *J. Phys. Condens. Matter* **2014**, 26, 183001.
- [81] J. Behler, *J. Chem. Phys.* **2011**, 134, 074106.
- [82] N. Artrith, T. Morawietz, J. Behler, *Phys. Rev. B* **2011**, 83, 153101.
- [83] T. Morawietz, V. Sharma, J. Behler, *J. Chem. Phys.* **2012**, 136, 064103.
- [84] S. Houlding, S. Y. Liem, P. L. A. Popelier, *Int. J. Quantum Chem.* **2007**, 107, 2817–2827.
- [85] C. M. Handley, P. L. A. Popelier, *J. Chem. Theory Comput.* **2009**, 5, 1474–1489.
- [86] S. A. Ghasemi, A. Hofstetter, S. Saha, S. Goedecker, *Phys. Rev. B* **2015**, 92, 045131.
- [87] S. Faraji, S. A. Ghasemi, S. Rostami, R. Rasoulkhani, B. Schaefer, S. Goedecker, M. Amsler, *Phys. Rev. B* **2017**, 95, 104105.
- [88] A. P. Bartók, G. Csányi, *Int. J. Quantum Chem.* **2015**, 115, 1051–1057.
- [89] M. Rupp, *Int. J. Quantum Chem.* **2015**, 115, 1058–1073.
- [90] A. P. Bartók, R. Kondor, G. Csányi, *Phys. Rev. B* **2013**, 87, 184115.
- [91] K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O. A. von Lilienfeld, K.-R. Müller, A. Tkatchenko, *J. Phys. Chem. Lett.* **2015**, 6, 2326–2331.
- [92] P. E. Dolgirev, I. A. Kruglov, A. R. Oganov, *AIP Adv.* **2016**, 6, 085318.
- [93] O. A. von Lilienfeld, R. Ramakrishnan, M. Rupp, A. Knoll, *Int. J. Quantum Chem.* **2015**, 115, 1084–1093.
- [94] P. Geiger, C. Dellago, *J. Chem. Phys.* **2013**, 139, 164105.
- [95] S. S. Schoenholz, E. D. Cubuk, D. M. Sussman, E. Kaxiras, A. J. Liu, *Nat. Phys.* **2016**, 12, 469–471.
- [96] E. D. Cubuk, S. S. Schoenholz, J. M. Rieser, B. D. Malone, J. Rottler, D. J. Durian, E. Kaxiras, A. J. Liu, *Phys. Rev. Lett.* **2015**, 114, 108001.
- [97] B. Jiang, H. Guo, *J. Chem. Phys.* **2013**, 139, 054112.
- [98] B. Jiang, J. Li, H. Guo, *Int. Rev. Phys. Chem.* **2016**, 35, 479–506.
- [99] L. B. Pártay, A. P. Bartók, G. Csányi, *J. Phys. Chem. B* **2010**, 114, 10502.
- [100] K. Toyoura, D. Hirano, A. Seko, M. Shiga, A. Kuwabara, M. Karasuyama, K. Shitara, I. Takeuchi, *Phys. Rev. B* **2016**, 93, 054112.
- [101] M. Hellström, J. Behler, *J. Phys. Chem. Lett.* **2016**, 7, 3302–3306.
- [102] V. Quaranta, M. Hellström, J. Behler, *J. Phys. Chem. Lett.* **2017**, 8, 1476.
- [103] T. Morawietz, J. Behler, *J. Phys. Chem. A* **2013**, 117, 7356.
- [104] S. Chiriki, S. S. Bulusu, *Chem. Phys. Lett.* **2016**, 652, 130–135.
- [105] S. Chiriki, S. Jindal, S. S. Bulusu, *J. Chem. Phys.* **2017**, 146, 084314.
- [106] B. Cheng, J. Behler, M. Ceriotti, *J. Phys. Chem. Lett.* **2016**, 7, 2210–2215.
- [107] M. Hellström, J. Behler, *Phys. Chem. Chem. Phys.* **2017**, 19, 82.
- [108] S. K. Natarajan, J. Behler, *Phys. Chem. Chem. Phys.* **2016**, 18, 28704.
- [109] L. Shen, J. Wu, W. Yang, *J. Chem. Theory Comput.* **2016**, 35, 479–506.
- [110] J. Behler, R. Martoňák, D. Donadio, M. Parrinello, *Phys. Status Solidi B* **2008**, 245, 2618–2629.
- [111] R. Z. Khaliullin, H. Eshet, T. D. Kühne, J. Behler, M. Parrinello, *Nat. Mater.* **2011**, 10, 693–697.
- [112] H. Eshet, R. Z. Khaliullin, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **2010**, 81, 184107.
- [113] H. Eshet, R. Z. Khaliullin, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. Lett.* **2012**, 108, 115701.
- [114] G. C. Sosso, G. Miceli, S. Caravati, J. Behler, M. Bernasconi, *Phys. Rev. B* **2012**, 85, 174013.
- [115] G. C. Sosso, D. Donadio, S. Caravati, J. Behler, M. Bernasconi, *Phys. Rev. B* **2012**, 86, 104301.
- [116] G. C. Sosso, J. Behler, M. Bernasconi, *Phys. Status Solidi B* **2012**, 249, 1880–1885.
- [117] G. C. Sosso, G. Miceli, S. Caravati, F. Giberti, J. Behler, M. Bernasconi, *J. Phys. Chem. Lett.* **2013**, 4, 4241–4246.
- [118] K. V. J. Jose, N. Artrith, J. Behler, *J. Chem. Phys.* **2012**, 136, 194111.
- [119] T. Morawietz, J. Behler, *Z. Phys. Chem.* **2013**, 227, 1559–1581.
- [120] S. K. Natarajan, T. Morawietz, J. Behler, *Phys. Chem. Chem. Phys.* **2015**, 17, 8356.
- [121] N. Artrith, B. Hiller, J. Behler, *Phys. Status Solidi B* **2013**, 250, 1191–1203.
- [122] N. Artrith, A. M. Kolpak, *Nano Lett.* **2014**, 14, 2670–2676.
- [123] N. Artrith, A. M. Kolpak, *Comput. Mater. Sci.* **2015**, 110, 20.
- [124] V. Kapil, J. Behler, M. Ceriotti, *J. Chem. Phys.* **2016**, 145, 234103.
- [125] J. S. Elias, N. Artrith, M. Bugnet, L. Giordano, G. A. Botton, A. M. Kolpak, Y. Shao-Horn, *ACS Catal.* **2016**, 6, 1675–1679.
- [126] S. K. Natarajan, J. Behler, *Phys. Chem. Chem. Phys.* **2016**, 18, 28704.
- [127] M. Gastegger, J. Behler, P. Marquetand, *J. Chem. Phys.* **2016**, 144, 194110.
- [128] A. A. Peterson, *J. Chem. Phys.* **2016**, 145, 074106.
- [129] D. Lu, J. Qi, M. Yang, J. Behler, H. Song, J. Li, *Phys. Chem. Chem. Phys.* **2016**, 18, 29113.
- [130] B. Kolb, B. Zhao, J. Li, B. Jiang, H. Guo, *J. Chem. Phys.* **2016**, 144, 224103.
- [131] N. Artrith, A. Urban, *Comput. Mater. Sci.* **2016**, 114, 135–150.
- [132] J. R. Boes, M. C. Groenenboom, J. A. Keith, J. R. Kitchin, *Int. J. Quantum Chem.* **2016**, 116, 979–987.
- [133] S. Hajinazar, J. Shao, A. N. Kolmogorov, *Phys. Rev. B* **2017**, 95, 014114.
- [134] B. Kolb, X. Luo, X. Zhou, B. Jiang, H. Guo, *J. Phys. Chem. Lett.* **2017**, 8, 666–672.
- [135] J. R. Boes, J. R. Kitchin, *J. Phys. Chem. C* **2017**, 121, 3479–3487.
- [136] J. R. Boes, J. R. Kitchin, *Mol. Simul.* **2017**, 43, 346–354.
- [137] J. Cuny, Y. Xie, C. J. Pickard, A. A. Hassanali, *J. Chem. Theory Comput.* **2016**, 12, 765.
- [138] J. Sun, J. Wu, T. Song, L. Hu, K. Shan, G. Chen, *J. Phys. Chem. A* **2014**, 118, 9120–9131.
- [139] K. Yao, J. E. Herr, J. Parkhill, *J. Chem. Phys.* **2017**, 146, 014106.

Manuscript received: March 25, 2017

Accepted manuscript online: May 18, 2017

Version of record online: August 18, 2017