International Journal of QUANTUM CHEMISTRY

Constructing High-Dimensional Neural Network Potentials: A Tutorial Review†

Jörg Behler

A lot of progress has been made in recent years in the development of atomistic potentials using machine learning (ML) techniques. In contrast to most conventional potentials, which are based on physical approximations and simplifications to derive an analytic functional relation between the atomic configuration and the potential-energy, ML potentials rely on simple but very flexible mathematical terms without a direct physical meaning. Instead, in case of ML potentials the topology of the potential-energy surface is "learned" by adjusting a number of parameters with the aim to reproduce a set of reference electronic structure data as accurately as possible. Due to this bias-free construction, they are applicable to a wide range of systems without changes in their functional form, and a very high accuracy close to the underlying first-principles data can be obtained. Neural network potentials (NNPs), which have first been proposed about two decades ago, are an important class of ML potentials. Although the first NNPs have been restricted to small molecules with only a few degrees of freedom, they are now applicable to high-dimensional systems containing thousands of atoms, which enables addressing a variety of problems in chemistry, physics, and materials science. In this tutorial review, the basic ideas of NNPs are presented with a special focus on developing NNPs for high-dimensional condensed systems. A recipe for the construction of these potentials is given and remaining limitations of the method are discussed. © 2015 Wiley Periodicals, Inc.

DOI: 10.1002/qua.24890

Introduction

An accurate description of the atomic interactions is of vital importance for carrying out reliable computer simulations in chemistry, physics, and materials science. Using the Born-Oppenheimer approximation,[1] various electronic structure methods are available to directly calculate the potential-energy and the atomic forces for a given nuclear configuration based on the laws of quantum mechanics. Often, the potential-energy of interest is the ground state energy, although the methods discussed below can equally be applied to excited state energies if they are available. The optimum choice of the electronic structure method depends on the system and typically requires to find an acceptable compromise between efficiency and accuracy for the problem of interest, as the exact solution of the Schrödinger equation is impossible for essentially all relevant problems. Each of these electronic structure calculations then provides a specific point on the multidimensional potential-energy surface (PES), which is in general a real-valued function depending on all atomic coordinates in the system yielding its potential-energy. In this review, the terms PES and "potential" will be used synonymously for this function.

For practical reasons, the number of electronic structure energies that can be calculated and stored is limited. Therefore, even in large data bases the energies and forces of most configurations that are visited, for example, in molecular dynamics (MD) simulations, will not be available beforehand. Consequently, in *ab initio* MD[2,3] the energies and forces need to be calculated "on-the-fly", typically using density-functional theory (DFT).[4,5] Alternatively, an analytic expression for the PES can be constructed and used in the simulations, which allows to perform MD simulations more efficiently as the evaluation of such expressions is much faster than solving the quantum mechanical problem. Also this approach is justified by the Born-Oppenheimer approximation, because the Hamiltonian and, therefore, also the ground state potential-energy is completely defined by the atomic positions, the nuclear charges, and the total charge of the system. Consequently, in principle a well-defined relation between the atomic structure and its potential-energy exists. Unfortunately, in most cases the corresponding functional form is too complicated to be derived analytically.

A pragmatic solution to this problem is the introduction of approximate PESs, and there are two fundamental approaches. In the conventional approach, the solution of the Schrödinger equation is replaced using a simplified energy expression based on physical considerations and reasonable approximations. The accuracy of these "physical potentials," which represent the vast majority of potentials in the literature, is thus

Jörg Behler
Lehrstuhl für Theoretische Chemie, Ruhr-Universität Bochum, D-44780, Bochum, Germany
E-mail: joerg.behler@theochem.ruhr-uni-bochum.de

†This tutorial review has been presented as a lecture at the "Jülich School on Computational Trends in Solvation and Transport in Liquids" taking place March 23–27, 2015, at the Jülich Supercomputing Center, Forschungszentrum Jülich, Germany. A closely related version of this review titled "High-Dimensional Neural Network Potentials as a Tool to Study Solvation" is included in the lecture notes of this School edited by Godehard Sutmann, Johannes Grotendorst, Gerhard Gompper, and Dominik Marx and is available at http://www.fz-juelich.de/conferences/STL-2015/EN/Home/home_node.html.

Jörg Behler is the head of an Independent Junior Research Group at the Lehrstuhl für Theoretische Chemie, Ruhr-Universität Bochum, Germany. He studied chemistry at the University of Dortmund. In 2000 he joined the group of Matthias Scheffler at the Fritz-Haber Institut der Max-Planck Gesellschaft in Berlin, where he obtained his PhD in 2004. For his work on the interaction of oxygen molecules with aluminium surfaces he received the Otto-Hahn medal of the Max-Planck Gesellschaft. After a postdoctoral stay with Karsten Reuter at the same institute, in 2006 he moved to the group of Michele Parrinello at the ETH Zürich. In 2007, he then established his own group in Bochum funded by a Liebig fellowship of the Fonds der Chemischen Industrie, the Emmy Noether program of the DFG and a Heisenberg fellowship. In 2013, he received the Hans G.A. Hellmann award for Theoretical Chemistry of the Arbeitsgemeinschaft Theoretische Chemie. His main research interest is the development of accurate and efficient atomistic potentials using artificial neural networks and electronic structure calculations. These potentials are then applied to problems in materials science as well as to chemical processes at interfaces and in solution.

limited by the imposed fixed functional form. Still, in most cases the essential features of the PES are described correctly, and reliable simulations can be carried out. There are many different physical potentials of varying form and complexity, ranging from classical force fields[6–10] in biochemistry to reactive potentials in materials science.[11–13]

The central idea of classical force fields is the decomposition of the total energy into low-dimensional bonding two-, three-, and four-body terms representing covalent bonds, bonding angles, and dihedral angles. In addition, electrostatic and van der Waals interactions given by Coulombs law and the Lennard-Jones potential, respectively, are used. In the low-dimensional bonding terms, only the immediate environment is taken into account. Since this approach would not be feasible based on the specification of the chemical elements alone, as the bonding properties are strongly influenced by the positions of the neighboring atoms, additional information on the bonding properties of the atoms is included via the introduction of atom types classifying the atoms according to functional groups. In this way, total energy expressions with low-dimensional, approximately additive energy terms can be obtained. The most severe limitation of classical force fields, that is, their inability to describe the making and breaking of bonds, has been overcome in several reactive force fields, for example, in the ReaxFF method.[14]

Most atomistic potentials in the field of materials science are based on the concept of the bond order[15] to take into account the effect of the atomic environments on the bonding properties. This is of fundamental importance, as in many problems of materials science involving systems like metals and alloys a good description of very different atomic environments is mandatory due to the importance of many-body contributions to the potential-energy. These render a decomposition of the potential into individual low-dimensional terms impossible. Further, simulations of these systems involve significant atomic rearrangements and thus require reactive potentials without the definition of fixed atom types.

An alternative class of atomistic potentials employs very flexible functions without a direct physical meaning. The aim of these "mathematical potentials" or machine learning (ML) potentials is to fit an analytic expression to a set of reference data obtained in electronic structure calculations as accurately as possible. These potentials are not yet widely distributed, but a number of promising approaches have been proposed in recent years differing in the types of functions that are used. They comprise permutation invariant polynomials,[16,17] the modified Shepard method using Taylor expansions,[18,19] Gaussian processes,[20–24] interpolating moving least squares,[25,26] artificial neural networks (NN),[27,28] and also support vector machines.[29] A more detailed discussion and comparison of these approaches can be found elsewhere.[30] Although the construction and use of these potentials has to be done with care to ensure that the correct physical shape of the PES is obtained, they are numerically extremely accurate and can be constructed even in difficult cases, for which no reasonable approximate physical potential can be found.

Physical as well as mathematical potentials provide the energy directly as a function of the atomic positions using well-defined analytic functions. Therefore, they both represent "atomistic potentials", and using such potentials enables the calculation of energies and forces many orders of magnitude faster than by applying electronic structure methods. They are particularly useful

- if long MD trajectories or extended Monte Carlo simulations are required
- if many MD trajectories are needed to obtain statistically converged results

and/or

- if the systems are too large for the application of electronic structure methods

In all these cases, they allow to extend the time and length scales of computer simulations beyond the realm of *ab initio* methods provided that a suitable functional form and a reliable set of parameters can be found, which is often a substantial challenge.

Mathematical potentials like the neural network potentials (NNPs) discussed in this review are capable of describing even very complex PESs, for which physical potentials are difficult

or even impossible to derive. Typical candidates for applications of such potentials are systems containing

- many different types of interactions, like covalent, ionic, and metallic bonding as well as dispersion interactions.
- "unusual" atomic environments, for example, in amorphous systems, structures emerging during phase transitions or being present in coordination chemistry.
- complicated reaction and transition paths involving the making and breaking of bonds.

Artificial NNs[31,32] have been introduced in 1943 to model and understand the signal processing in the brain,[33] and in the following decades they have found wide use in many fields of science[34] due to their pattern recognition and data classification capabilities. There are many types of NNs with different functional forms.[31,32] A general definition covering all these types has been given by Kohonen [35]:

"Artificial neural networks are massively parallel interconnected networks of simple (usually adaptive) elements and their hierarchical organizations which are intended to interact with the objects of the real world in the same way as biological nervous systems do."

In particular, multilayer feed-forward (MLFF) NNs have been demonstrated to be useful for the construction of PESs due to their ability to represent arbitrary functions. It has been proven independently by a number of researchers that MLFF NNs are "universal approximators",[36–40] that is, they enable approximating unknown multidimensional functions to an, in principle, arbitrary accuracy based on a set of known function values:

"[. . .] standard multilayer feedforward networks with as few as one hidden layer using arbitrary squashing functions are capable of approximating any [. . .] function from one finite dimensional space to another to any desired degree of accuracy, provided sufficiently many hidden units are available. In this sense, multilayer feedforward networks are a class of universal approximators."[37]

This is the theoretical foundation for using NNs to construct atomistic potentials. However, in practice this formal proof is of limited use as neither the number or distribution of training points nor the required size of the NN for the representation of these points is known. Still, the remarkable result is that there is no fundamental restriction in the accuracy that can be achieved when constructing NNPs, which is an important difference to physical potentials having intrinsic limitations due to their rather inflexible functional forms.

In general, NNPs can be defined by three criteria: [41]

1. NNPs provide a direct functional relation between the atomic configuration and the potential-energy employing one or more artificial NNs.
2. NNPs are constructed using a set of first-principles data, usually total energies and sometimes energy derivatives, obtained from a single electronic structure method. Experimental data cannot be used as experiments often have uncertainties being substantially larger than the fitting errors of NNPs. For the same reason, results from different electronic structure methods cannot be combined as numerically inconsistent information must be avoided.
3. NNPs do not contain any approximations apart from the intrinsic limitations of the chosen reference electronic structure method. No other *ad hoc* empirical functional components are included.

NNPs offer a number of advantages for the construction of PESs:

- Energies can be fitted to high accuracy with very small remaining errors compared to the underlying reference data.
- NNPs can be calculated efficiently and require much less CPU time than electronic structure calculations.
- No knowledge about the functional form of the PES is required.
- The NN energy expression is unbiased, generally applicable to all types of bonding and does not require system-specific modifications.

Still, there are also disadvantages of NNPs that one should be aware of:

- The evaluation of NNPs is notably slower than the use of simple classical force fields.
- NNPs have no physical basis and only very limited extrapolation capabilities. Therefore, they can fail spectacularly if they are not used properly.
- The construction of NNPs requires substantial effort, and a large number of training points from electronic structure calculations is required.
- Currently, NNPs are limited to systems containing either only a few different chemical elements but many atoms or a small number of atoms with arbitrary nuclear charges.

In the past two decades, NNPs have been constructed for many types of systems. Initially, they have been restricted to small molecules[42–46] and systems, whose complexity has been strongly reduced by freezing a majority of the degrees of freedom, like small molecules interacting with frozen metal surfaces,[27,28,47–58] or extensions of water potentials to include polarization effects truncated at low order while keeping the water monomer geometries fixed.[59,60] A comprehensive summary of previous work on NNPs can be found in two recent reviews.[61,62]

Only a few attempts have been made to date to construct NNPs for high-dimensional systems. The term "high-dimensional" has been used frequently and with different meanings in the literature. Here, we consider a potential as being high-dimensional if it is applicable to systems containing thousands of atoms and all their degrees of freedom explicitly. One early approach to construct high-dimensional NNPs by Smith and coworkers published in 1999[63,64] is based on a decomposition of the system into chains of atoms of increasing length, whose energies are constructed using a

NN of variable size. Surprisingly, this promising idea has not been further developed until 2007, when it was extended to represent the PES of silicon using tight binding reference data.[65] Since then, the only application of this method has been the construction of the DFT PES of the same system in 2008.[66] Another very appealing approach introduced by Manzhos and Carrington is based on a rigorous many-body expansion of the potential-energy using redundant coordinates and a high-dimensional model representation.[67–71] Here, in essence, the potential-energy is expressed as a sum of terms of increasing order in the spirit of a many-body expansion, and it has been demonstrated that formally low-dimensional NNs can be used if the input coordinates are constructed as a function of the original atomic coordinates. This method, which is very accurate and has no fundamental conceptual limitations, has been applied to date only to rather small molecules, as the number of NN evaluations increases rapidly with system size making it very costly for larger systems. Still, it is probably the most systematic approach to construct NNPs.

In this tutorial review, the basic methodology of a NNP approach, which has been introduced by Behler and Parrinello[41,72,73] to address high-dimensional systems, and its extensions will be discussed with a focus on the construction and applicability. To date NNPs based on this approach have been constructed for various systems including silicon,[72,74,75] copper,[76] carbon,[77,78] sodium,[79,80] zinc oxide,[81] copper clusters supported at zinc oxide,[82] neutral[83–85] as well as protonated water clusters,[86] and the phase change material GeTe.[87–90] Many of the techniques described below are very general and can be applied to all types of NNPs and even beyond to other types of ML potentials.

Conventional NNPs

Most conventional NNPs use a single MLFF NN to construct a direct functional relation between the atomic configuration and the potential-energy. For this purpose, a number of artificial neurons is organized in several layers as shown in Figure 1. The potential energy E is obtained in the neuron, or node, in the output layer. It depends on the atomic configuration that is provided to the NN in form of a vector of input coordinates $\mathbf{G}=\{G_i\}$. The specific choice of input coordinates is a crucial aspect of any NNP, as these coordinates need to fulfill a number of requirements. In between the input and the output layer there are one or more so-called "hidden layers." As their name implies, they have no physical meaning but define the functional form of the NN and provide the required mathematical flexibility to construct the functional relation between the input and the output of the NN. The more hidden layers are used and the more nodes are included in each hidden layer, the more flexible is the functional form of the NN. A typical NN architecture contains two to three hidden layers and up to typically about 50 nodes per layer. The entity of all layers including the input and output layer as well as the number of nodes per layer defines the architecture of the NN. For instance, the NN shown in Figure 1 can be described by the

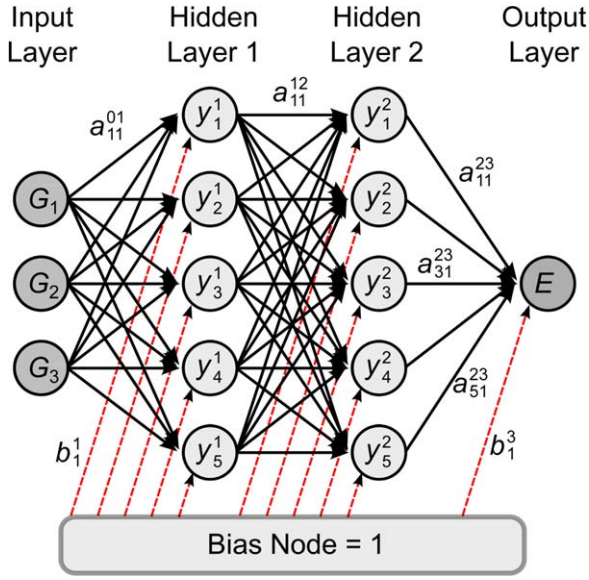

Figure 1. Schematic structure of a small feed-forward NN. The nodes are arranged in layers and the goal is to establish a functional relation between the energy output E and the structure described by a vector of input coordinates $\{G_i\}$. The functional form of the NN is given by eq. (2) (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

short notation 3–5-5-1, where each number specifies the number of nodes in one layer starting from the input layer.

As indicated by the arrows in Figure 1 each node in each layer is connected to the nodes in the adjacent layers by a weight parameter, where the symbol a_{ij}^{kl} is used for the parameter connecting node i in layer k with node j in layer l with $l=k+1$. The input layer has the superscript 0 and the arrows label the direction of the flow of information through the network. Further, there is a bias node connected to all nodes in the hidden layers and to the output node providing the number "one", which can be scaled by a bias weight b_i^j, where j is the layer of the target node and i is its number within this layer. The value y_i^j of any node i in any hidden layer j can then be calculated from the values of the N_{j-1} nodes in the previous layer and the connecting and bias weight parameters according to

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

This equation is essentially a linear combination of the values of the nodes in the previous layer, which has been shifted by the bias weight. As the topology of a PES generally cannot be expressed as a simple linear combination, a nonlinear function f_i^j is applied, which is called "activation function" or "basis function" of the NN. It has the purpose to provide the capability of fitting arbitrary functions. As part of the NN total energy expression the activation functions must be differentiable to determine analytic forces and to calculate the derivatives of the NN output with respect to the weight parameters needed for the gradient-based optimization of

the weights. In case that forces are used for the NN training as well,[76,91,92] also second derivatives of the activation functions are needed.

There are many possible choices for the activation functions. They all have in common that they have a nonlinear region and that they typically converge to 1, −1, or 0 for very large and very small arguments. Frequent choices are the hyperbolic tangent, the sigmoid function and Gaussians. Still, also periodic activation functions, for example, trigonometric functions,[93] or exponential functions[68] have been proposed. For the node in the output layer, often a linear function is used to avoid any restriction in the range of possible output values of the NN. The complete functional form of the small example NN in Figure 1 is a hierarchy of nested activation functions acting on linear combinations of the values in the previous layer given by

$$E=f_1^3\left(b_1^3+\sum_{k=1}^{5}a_{k1}^{23}\cdot f_k^2\left(b_k^2+\sum_{j=1}^{5}a_{jk}^{12}\cdot f_j^1\left(b_j^1+\sum_{i=1}^{3}a_{ij}^{01}\cdot G_i\right)\right)\right) \quad . \tag{2}$$

Starting from the left side of Figure 1, first the values of the nodes in the first hidden layer are calculated, followed by the nodes in the second hidden layer and so forth until the output node is reached. The NN energy expression also provides access to forces and higher derivatives. These should not be fitted separately using additional output nodes as this necessarily would result in numerical inconsistencies between the energies and forces. Instead the forces can be calculated from analytic derivatives as discussed below.

To obtain physically meaningful energy outputs, NNPs need to be trained to a set of known reference points obtained from electronic structure calculations. This is done by optimizing the weight parameters, which are initially chosen as random numbers, iteratively to minimize the error of this training set. This is often a very time-consuming step, as the training data sets can become very large and also the number of weight parameters N_w to be optimized can be substantial,

$$N_w=\sum_{k=1}^{M_{HL}+1}\left(N_{k-1}\cdot N_k+N_k\right) \quad , \tag{3}$$

where M_{HL} is the number of hidden layers and N_k is the number of nodes in layer k including the output layer. NNPs containing only a single feed-forward NN have been constructed successfully for a number of low-dimensional systems containing up to about 12 degrees of freedom. A highly recommended tutorial-style discussion of fitting simple model functions by NNs can be found elsewhere.[91]

High-Dimensional NNPs

Using a single feed-forward NN for the construction of high-dimensional NNPs is impossible for several reasons. First of all, for each atom added to the system three new degrees of freedom need to be included in form of additional input nodes. If the total number of input nodes becomes too large, the construction of the NNP will be inefficient because there are too many weight parameters to be determined in the fitting process. Further, in applications of the NNP the calculation of the output energy will be slow. Finally, constructing a sufficiently dense set of training points becomes very challenging in a too high-dimensional coordinate space both in terms of selecting the points and in terms of CPU time.

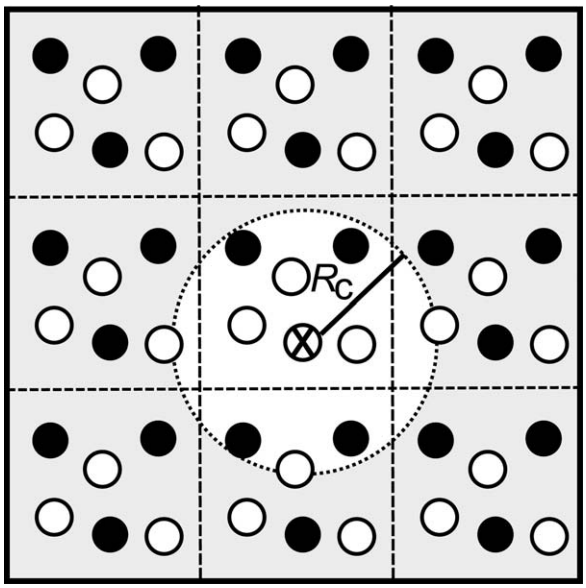

Figure 2. Illustration of the atomic environment defined by a cutoff radius R_c in a periodic system containing two different elements (white and black circles). In high-dimensional NNPs, all atoms inside the cutoff sphere determine the energy contribution of the central atom to the potential-energy (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

Another fundamental problem, which is present even for small molecules, is related to the symmetry of the NN. For a water monomer, which can be described by three interatomic distances, both OH bonds are chemically equivalent. If the positions of both hydrogen atoms are exchanged, the atomic configuration must still have the same potential-energy since chemically the structure has not changed. As the connecting weights of a NNP all have numerically different values, changing the order of the input coordinates, that is, the sequence of the two OH-bond lenghts, will result in a numerically different energy output. This problem has been recognized very early, and a solution has been proposed already in 1998 by Gassner et al.[60] Instead of using a set of interatomic distances a symmetrization step corresponding to a transformation of the initial internal coordinates is introduced in this approach. Consequently, chemically equivalent atoms lose their identity and all possible representations, that is, orders of the atoms, yield the same NN input vector. Therefore, the same output energy is obtained as it should be. Unfortunately, this method, which has also been extended to the scattering of molecules at solid surfaces,[48] is applicable only to very small systems, as the complexity of the symmetrization step increases rapidly with the number of atoms. Even this simple example illustrates the crucial importance of the choice of input coordinates, and this topic will be discussed in more detail below.

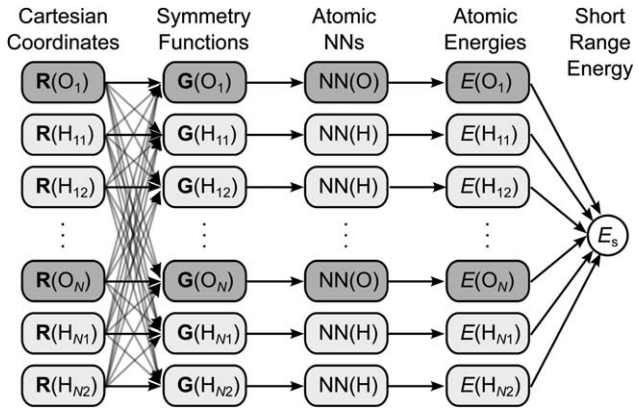

Figure 3. Schematic structure of a high-dimensional NNP for a system containing N water molecules. First, for each atom the Cartesian coordinate vectors R are transformed to symmetry function vectors G, which describe the atomic environments and serve as input for the atomic NNs providing the atomic energies. Finally, the atomic energy contributions are added to yield the short range energ E_s (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

Finally, any NNP consisting of a single NN is only applicable to the system size that has been used for its construction, which is another severe problem. If an atom would be added when applying the potential, the connecting weights for the additional input nodes would not be available. If, conversely, an atom would be removed, its NN input coordinates would be ill-defined, and using simply a set of zero input values for these nodes is not possible because these numbers would change in frequently used preprocessing steps of the input coordinates (cf. below).

A solution to all these problems is to construct the energy E_s of the system as a sum of N_{atom} atomic energy contributions E_i, which are provided by a set of individual atomic NNs,

$$E_s = \sum_{i=1}^{N_{atom}} E_i. \quad (4)$$

The atomic energy contributions depend on the local chemical environments up to a cutoff radius R_c as shown in Figure 2. This cutoff radius, which typically has a value between 6 and 10 Å, is a convergence parameter and needs to be tested to ensure that all energetically relevant interactions are included. The positions of the neighboring atoms within the resulting cutoff sphere are then described by a set of many-body symmetry functions. They serve as a structural fingerprint and are supplied to the atomic NNs as vectors containing usually between 20 and 100 function values. The structure of the atomic NNs is essentially identical to the structure of the single-NN NNPs discussed in the previous Section (cf. Fig. 1) with the only difference that the output is now just a part of the total energy.

The resulting high-dimensional NN approach is shown schematically in Figure 3 for a system consisting of N water molecules. For each atom, there is a separate line starting from the Cartesian coordinate vector **R** of the atom. In the next step, a vector of many-body symmetry functions **G** is constructed for each atom, which describes the arrangement of all atoms in the chemical environment. Consequently, the symmetry function values depend on the Cartesian coordinates of all other atoms as indicated by the grey arrows. This vector of symmetry function values is the input for the atomic NN yielding the energy contribution of the atom. Once each atomic energy contribution has been determined, the total energy E_s is obtained by summing all contributions.

It should be noted that although the atomic NNs provide "atomic energies", no information about partitioned atomic energies is required and the weight parameters can be determined using total energies from electronic structure calculations only. The condition of permutation invariance of the total energy with respect to the order of the atoms is automatically satisfied in this method as the order of the summation does not change the resulting total energy. For each element, like oxygen or hydrogen, there is a separate type of atomic NN with its own architecture and set of weight parameters, but for a given element all atomic NNs are identical. Different atomic energy contributions arise only because of the different NN input vectors reflecting the different environments. As a result, the high-dimensional NN method is also applicable to systems of different size, in the training of the NNP as well as in its application. If an atom is added, the scheme in Figure 3 is extended by another line of the respective element. If an atom is removed from the system, its line is deleted.

In summary, using the energy expression in eq. (4) the most severe problems of conventional NNPs can be solved:

- The introduction of a cutoff reduces the effective dimensionality to the energetically relevant interactions, which allows to use NNs of tractable size. Still, the remaining atomic environments often contain up to 100 neighboring atoms or more enabling the inclusion of high-order many-body effects.
- The total energy is invariant with respect to the order of the atoms.
- The NNP is applicable to systems containing different numbers of atoms without the need to determine a different set of weight parameters for each system size.

In the original high-dimensional NNP method of Behler and Parrinello,[72] all interactions between atoms separated by more than the cutoff radius are truncated. Still, there might be interactions beyond this radius that should be included, like rather long-ranged electrostatic interactions. For this purpose, it is possible to extend the total energy expression in eq. (4) by an additional electrostatic term[81,83] E_{elec},

$$E = E_s + E_{elec} = \sum_{i=1}^{N_{atom}} E_i + E_{elec}. \quad (5)$$

The electrostatic energy is calculated using either Coulombs law or an Ewald summation,[94] that is, no spatial truncation is applied. The required charges are constructed as environment-dependent atom-centered charges expressed by a second set of atomic NNs in the same way as the (now "short-ranged") energy contributions in Figure 3. Like atomic energies, atomic charges are no quantum mechanical observables, but still applying

charge partitioning methods is much more common than using energy partitioning schemes.[95,96] We have therefore chosen to use Hirshfeld charges[97] obtained in electronic structure calculations to determine the weight parameters of the atomic charge NNs. In principle, however, it would also be possible to couple the fitting of the weight parameters of the short-range energy and the charge NNs and to determine the charges under the only constraint that the best possible total energy is obtained. This, however, is computationally substantially more demanding. Also higher electrostatic multipoles can be represented by NNs, as has been demonstrated by Popelier and co-workers.[98,99]

It should be noted, however, that according to our experience including long-range electrostatic interactions does only marginally improve the description of the PES for most systems as all electrostatic interactions within the cutoff region can be described as well by the short range part of the NNP, which does not distinguish between electrostatic and nonelectrostatic interactions. Consequently, the electrostatic NN is only relevant for electrostatic interactions of atoms separated by more than the cutoff radius. Including electrostatics explicitly requires to separate the reference total energies into an electrostatic and a short range contribution before the fitting to avoid a double counting of the electrostatic part. Because of the singularity of the Coulomb interaction at short distances, removing the electrostatic energy from the total energy can even give rise to an increased corrugation of the remaining short range part of the potential. This complicates the fitting and can be avoided by screening the electrostatic interactions at short distances.[83] This screening has no consequences for the accuracy of the potential as the screened part of the electrostatic energy is by definition compensated by the short-range energies.

Finally, also van der Waals terms can be added to NNPs. Conceptually, there is no need to add these terms as all kinds of interactions are treated by NNPs on an equal footing and the most relevant part of van der Waals interactions is included inside the cutoff radius. Still, DFT calculations are frequently used to train NNs and depending on the choice of the exchange correlation functional DFT often provides a poor description of van der Waals interactions, which is a well-known deficiency of GGA functionals that has received a lot of attention in recent years.[100] A number of correction schemes have been proposed to improve the description of van der Waals interactions. In case of NNPs in particular the D3 method by Grimme[101] has been found to be useful,[84] because the NN energies are numerically very close to the DFT energies. Consequently, the D3 correction can equally be applied to the NN energies after the evaluation of the NNP. Alternatively, the corrections can be applied to the training set before fitting the NN parameters, which reduces the computational costs of the NNP to that of standard high-dimensional NNPs. This also enables using other correction schemes.[102]

Symmetry Functions

Properties of symmetry functions

Having established the structure of high-dimensional NNPs, one of the most important decisions to be made concerns the description of the atomic configurations using a suitable set of coordinates. Cartesian coordinates cannot be used at all, because their numerical values are not invariant with respect to translations and rotations of the system. Since, conversely, the structure of a system and thus its energy remains unchanged upon these operations, so must be the set of input coordinates. In case of NNPs, many different types of coordinates have been used over the years like interatomic distances,[59] symmetrized distances,[60] functions including the symmetry of surfaces,[28,48] complex combinations of coordinates in a many-body expansion,[69] and many others.

In this review, the transformed coordinates, which are many-body functions of all atomic positions inside the cutoff spheres, will be called "symmetry functions" for historic reasons. The term "symmetry" does not refer to the point group or the space group of the system, but to the basic requirement that structurally equivalent representations of the system must give rise to the same set of coordinate values. The most common but also the most critical operation is the permutation of atoms, that is their order in the structural description, since the order of the atoms is arbitrary. Any exchange of atoms of the same element, which is not at all restricted to positions being equivalent by symmetry, must not affect the vector of symmetry function values. Constructing such permutation invariant input coordinates, which is a common problem for all ML potentials,[103] has been a major challenge in the advent of NNPs. Symmetries in the traditional meaning of symmetry operations do not play an important role, since they are included as a special case in the permutation symmetry and since in most applications like MD simulations at finite temperatures systems possess only C_1 symmetry.

For high-dimensional NNPs using a cutoff radius to define the energetically relevant atomic environment it must be ensured that the number of symmetry functions does not depend on the number of atoms inside the cutoff sphere, as this can change in the course of MD simulations. Another obvious criterion for the choice of the symmetry functions is their ability to distinguish different atomic environments and to provide similar coordinate values for similar atomic configurations to facilitate the fitting process. Contradictory data, which emerge if two different structures with different energies have closely resembling symmetry function values, must be avoided as this strongly hampers the determination of the NN weight parameters. Finally, we note that even complicated functional forms of the symmetry functions are acceptable, since the transformation of the coordinates does not need to be inverted. In both cases, in the training of the NNP as well as in its application, the transformation is starting from the Cartesian coordinates to obtain the symmetry functions. A transformation in the opposite direction is not required.

In summary, a set of symmetry functions for the construction of high-dimensional NNPs must have the following properties:

- rotational and translational invariance
- invariance with respect to the permutation of atoms of the same element
- provide a unique description of the atomic positions
- constant number of function values independent of the number of atoms in the cutoff spheres

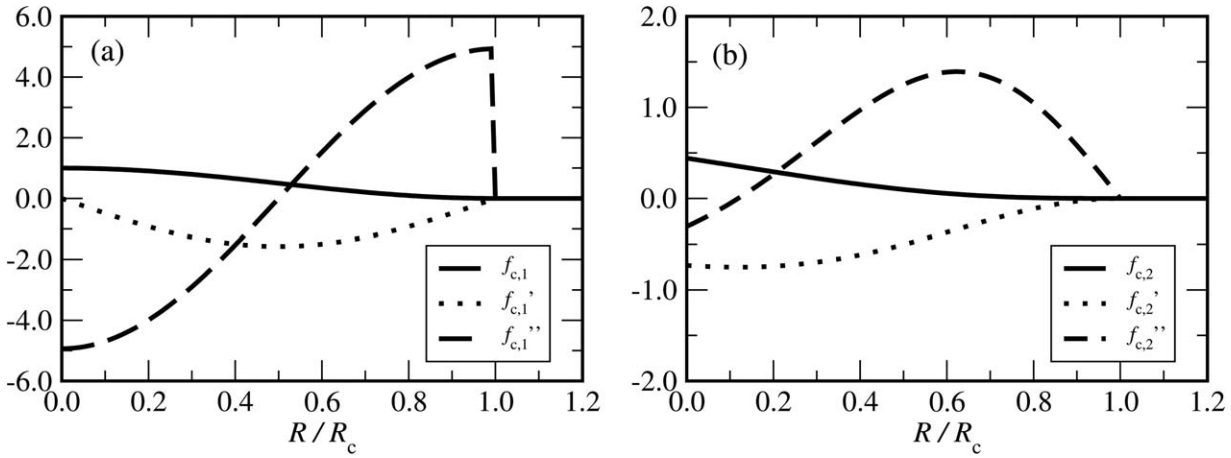

Figure 4. Plot of the cutoff functions $f_{c,1}$ and $f_{c,2}$ and their derivatives as defined in eqs. (6) and (7), respectively, (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

In the following section, the functional forms of several types of symmetry functions suitable for the construction of high-dimensional NNPs will be discussed. Further details and additional functions can be found elsewhere.[73]

The Functional Form of the Symmetry Functions

An important component of all symmetry functions is the cutoff function f_c defining the atomic environments. Two possible choices are the function

$$f_{c,1}(R_{ij}) = \begin{cases} 0.5 \cdot \left[\cos\left(\frac{\pi R_{ij}}{R_c}\right)+1\right] & \text{for} \quad R_{ij} \le R_c \\ 0.0 & \text{for} \quad R_{ij} > R_c \end{cases}, \tag{6}$$

which is essentially the monotonically decreasing part of a cosine function whose domain has been rescaled from π to R_c and the function

$$f_{c,2}(R_{ij}) = \begin{cases} \tanh^3\left[1-\frac{R_{ij}}{R_c}\right] & \text{for} \quad R_{ij} \le R_c \\ 0.0 & \text{for} \quad R_{ij} > R_c. \end{cases} \tag{7}$$

Both functions decrease with increasing distance R_{ij} between the central atom i and its neighbor j, which reflects qualitatively the reducing strength of the interactions between the atoms. At the cutoff radius R_c both functions have zero value and slope as shown in Figure 4. This is important for the calculation of the forces, which require the determination of the derivatives of the symmetry functions with respect to the atomic Cartesian coordinates. Additionally, in case of cutoff function $f_{c,2}(R_{ij})$ also the second derivative is zero at R_c. Formally, this is important to avoid discontinuities in the change of the forces when atoms enter or leave the cutoff spheres in MD simulations as this can possibly result in problems to achieve energy conservation. We found that as long as sufficiently large cutoff radii are used, that is, 6 Å or more, also function $f_{c,1}(R_{ij})$ provides an excellent energy conservation. In the symmetry functions discussed in the remaining part of this Section, the function f_c can be either of these two functions.

Based on the cutoff function as a central component, several types of many-body symmetry functions can be constructed. There are two classes of symmetry functions: "radial" symmetry functions, which describe the radial distribution of neighbors up to the cutoff radius, and "angular" symmetry functions specifying their angular arrangement. All symmetry functions have in common that each of them depends on the positions of all atoms inside the cutoff spheres. Therefore, in contrast to internal coordinates like bond lengths, their numerical values are not always easy to illustrate. Apart from facilitating the description of many-body contributions to the potential another important reason to use symmetry functions depending on all neighboring atoms is to obtain a constant number of function values independent of the number of atoms in the cutoff spheres. It is neither possible nor desirable to use a separate symmetry function for each neighboring atom, like a distance, because then the number of input nodes of the atomic NNs would change with the number of atoms in the cutoff spheres and the use of a single atomic NN per element would be impossible.

The most basic radial symmetry function describing the environment of atom i is simply a sum of the cutoff functions for all neighboring atoms j inside the cutoff sphere,

$$G_i^1 = \sum_{j=1}^{N_{atom}} f_c(R_{ij}). \tag{8}$$

Consequently, a single number is obtained, which can be interpreted as a coordination number up to the cutoff radius. This single number is of course not sufficient to describe the radial arrangement of the neighboring atoms. This can be achieved by using a set of these functions with different spatial extensions given by different cutoff radii as plotted in Figure 5a. Still, this function type needs to be used with care as too short cutoff radii can give rise to artefacts in the forces close to the cutoff radius. A detailed discussion of this aspect can be found elsewhere.[73]

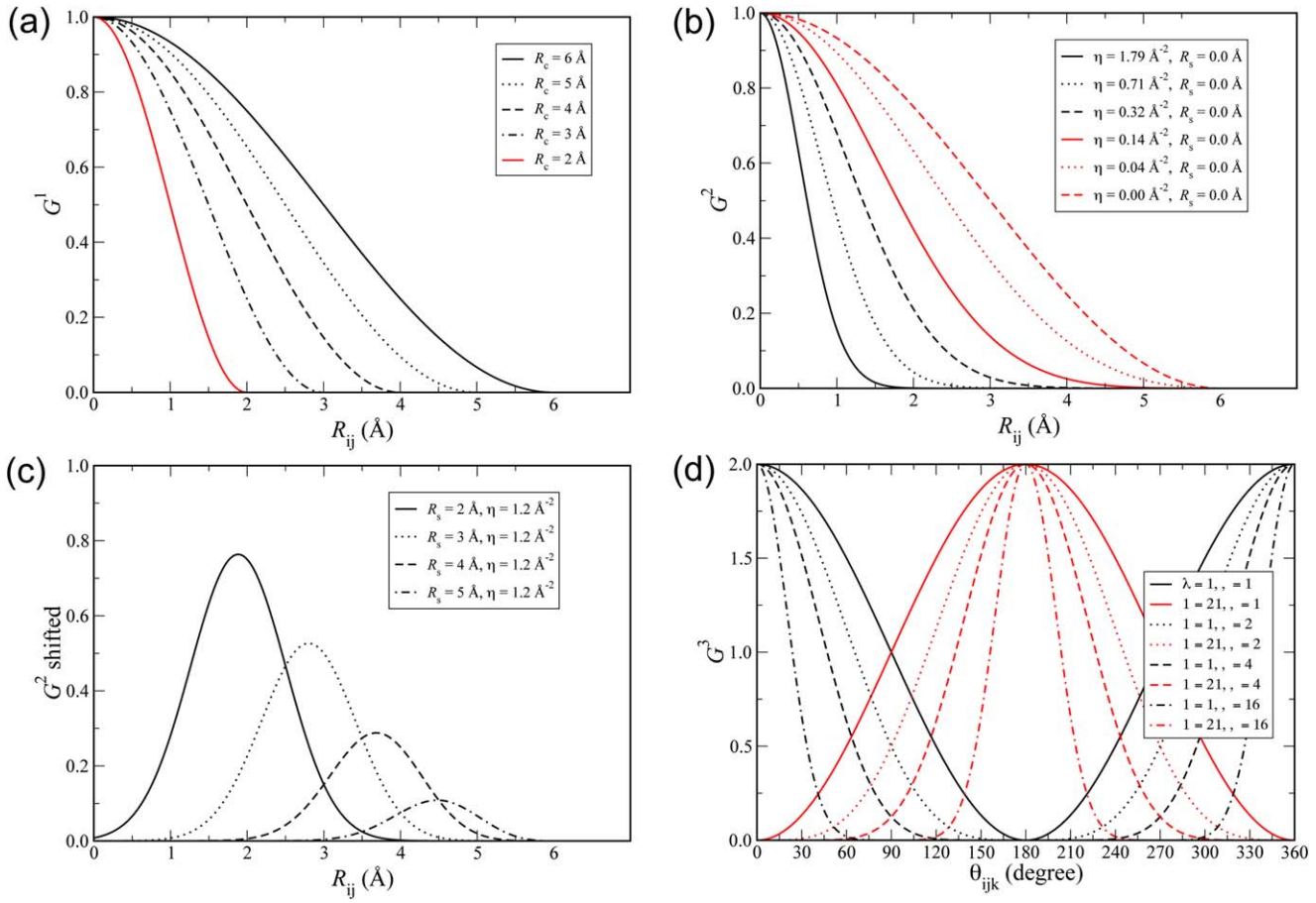

Figure 5. Plot of the radial symmetry functions $G^{[1]}$ and $G^{[2]}$ [panels (a)–(c)] and of the angular part of the angular symmetry function $G^{[3]}$ [panel (d)]. For $G^{[2]}$ a cutoff radius of 6 Å has been used. (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

A better choice to describe the radial arrangement of atoms in the cutoff sphere is a sum of products of Gaussians and the cutoff function,

$$G_i^2 = \sum_{j=1}^{N_{\text{atom}}} e^{-\eta\left(R_{ij}-R_s\right)^2} \cdot f_c\left(R_{ij}\right). \tag{9}$$

Here, for all functions the same large cutoff should be used to obtain reliable forces for all interatomic distances. Instead, the effective radial extension of the symmetry functions can be controlled by the parameter η, which defines the width of the Gaussians (cf. Fig. 5b). In this way, a strict decay to exactly zero in value and slope at too short radii can be avoided while still different radii can be probed by the symmetry functions using different values of η.[73] For nonzero values of the shifting parameter R_s the center of the Gaussians can also be displaced to certain radii resulting in spherical shells of a width given by η as shown in Figure 5c to improve the sensitivity of the symmetry functions at specific radii.

Describing the radial distribution of neighboring atoms is not sufficient to obtain a suitable structural fingerprint of the atomic environments. This can be achieved by using additional angular functions depending on the angle θ_{ijk}, which is centered at the central atom i and is enclosed by the two interatomic distances R_{ij} and R_{ik}. As the potential is periodic with respect to this angle, instead of θ_{ijk} its cosine is used. A suitable angular symmetry function can then be defined as a sum over all cosines with repect to any possible pair of neighbors j and k, which are multiplied by Gaussians of the three interatomic distances in the triplet of atoms and the respective cutoff functions,

$$G_i^3 = 2^{1-\zeta} \sum_{j\neq i} \sum_{k\neq i,j} \left[\left(1+\lambda\cdot\cos\theta_{ijk}\right)^{\zeta} \cdot e^{-\eta\left(R_{ij}^2+R_{ik}^2+R_{jk}^2\right)} \cdot f_c\left(R_{ij}\right) \cdot f_c\left(R_{ik}\right) \cdot f_c\left(R_{jk}\right)\right] \tag{10}$$

This functional form ensures that G_i^3 approaches zero if the distance between any two atoms in the triplet becomes larger than the cutoff radius, as in this case the triplet is not completely included within the cutoff sphere of all three atoms. Again, the width of the Gaussians η needs to be specified to take into account that the importance of the angle depends on the atomic separations. The distribution of angles can be probed by using different exponents ζ as shown for the plot of the angular part of G_i^3 in Figure 5d, and the normalization factor $2^{1-\zeta}$ ensures that the range of values is independent of the actual choice of exponent. The parameter λ, which can have values of $+1$ or -1 can be used to invert the shape of the cosine function to obtain a good description at

different values of θ_{ijk}. While for $\lambda = 1$ the maxima of the cosine terms are at $\theta_{ijk} = 0^\circ$, for $\lambda = -1$ they are located at $\theta_{ijk} = 180^\circ$.

An alternative angular function is given by

$$G_i^4 = 2^{1-\zeta} \sum_{j \neq i} \sum_{k \neq i,j} \left[\left(1 + \lambda \cdot \cos\theta_{ijk}\right)^\zeta \cdot e^{-\eta\left(R_{ij}^2 + R_{ik}^2\right)} \cdot f_c\left(R_{ij}\right) \cdot f_c\left(R_{ik}\right) \right]. \quad (11)$$

Here, the Gaussians and cutoff functions with respect to R_{jk} have been omitted. This modification has two consequences. First, the numerical values of G_i^4 are generally larger than of G_i^3, since $e^{-\eta R_{jk}^2}$ and $f_c(R_{jk})$ are both lower than one, making this function more useful for larger atomic separations. Second, it is possible to take into account atomic triplets in which j and k are both in the cutoff sphere of atom i but separated by more than R_c. As the maximum possible distance between j and k inside the cutoff sphere is $2 \cdot R_c$, a significantly larger number of angles is thus included in the summation resulting in very different values compared to angular function G_i^3. Further types of radial and angular symmetry functions and a discussion of their properties can be found elsewhere.[73] The ability of these symmetry functions to describe and distinguish different atomic environments can also be used for the classification of structures, as has been demonstrated by Geiger and Dellago.[104]

Several comments should be made on the definition of the symmetry function set. First, the parameters R_c, η, R_s, ζ, and λ define the spatial shape of the symmetry functions. Consequently, they are not optimized but remain fixed during the determination of the weights of the NN. In this sense, they are similar to basis functions in electronic structure calculations. We found that the specific values of the symmetry function parameters are surprisingly transferable from one system to another as the resulting symmetry functions even for very different systems have to cover a similar region in space. Second, as the NN weights have been obtained for a specific set of input symmetry functions, these functions need to be known when applying the NNP in simulations. Consequently, the symmetry functions can be considered as a part of the NNP. Finally, the number of symmetry functions grows rapidly with the number of elements in the system. For instance, for each value of η there is one function of type G^2 for each element in the neighborhood and for each element of the central atom. As the atomic NNs are separated and independent from each other, the different possible central elements are not relevant. Still, the combinatorial increase of the radial and in particular of the angular functions for a given central atom is currently limiting the number of elements that can be included in high-dimensional NNPs. It has also been demonstrated that high-dimensional NNPs can be constructed using environment-dependent atom-pair contributions to construct the potential-energy,[105] but this method is computationally more demanding than the atom-based approach, since the number of possible pairs is substantially larger.

Construction of Symmetry Function Sets

The choice of symmetry functions is a very important step for the construction of high-dimensional NNPs, as a reliable distinction of different structures is mandatory. If this distinction is not possible, for example, if too few symmetry functions are used, different reference structures with differing energies and forces give rise to the same set of symmetry function values. Consequently, these structures are identical for the NN, the PES is not defined as a unique function of the atomic configuration, and there may be different energies for "the same structure," which results in poor fits.

Although the construction of the symmetry function sets still is to some extent empirical, there is a number of analyses that can be carried out to investigate if a vector of symmetry functions is appropriate for the construction of a NNP:

- The symmetry functions should cover the configuration space in an unbiased way. A reasonable initial choice is to use, for example, an equidistant set of radial functions as shown in Figure 5b. The spatial extension of the function with the smallest effective range should be selected based on the shortest interatomic distances present in the data set. It must be avoided to include symmetry functions, which have a value of zero for all atoms and as a rule of thumb the turning point of the Gaussian with the largest η value should correspond to the shortest bond for the respective neighboring element. Consequently, for different neighboring elements, the symmetry function parameters may differ.
- For each symmetry function the range of values present in the data set should be analyzed. If the range of values, that is, the difference between the smallest and largest function value, is too small, the symmetry function is not contributing to the distinction of different structures and should be removed from the symmetry function set. Further, also the normalized range of values obtained through dividing by the standard deviation of the function values should be reasonable to ensure that the range of values is not dominated by a few outliers. If the difference between the symmetry function values is too small, the fits might become numerically unstable as the NN will try to assign substantial energy changes to tiny differences in the symmetry functions.
- To test if the symmetry functions allow to distinguish different structures, the data set should be investigated for contradictory data. In particular, the atomic forces represent a valuable test case, as they provide local atom-specific information about the PES. If, for example, the magnitude of the forces is very different for two atoms, they must have a different chemical environment, and consequently also the symmetry function vectors must be different. If the data set is searched systematically for pairs of atoms having similar symmetry function vectors but experiencing different forces, as discussed in Ref. [73], inappropriate symmetry function sets can be identified and augmented by further functions until different atomic environments can be distinguished.
- The set of symmetry functions should be kept as small as possible to increase the efficiency of the calculation of the NN energy output. This can be investigated by determining the correlation between the values of a given symmetry function for all atoms in the reference set. If there is

a high correlation, the symmetry functions are (close to) linearly dependent and essentially no additional information is provided by using both instead of only one of these symmetry functions. This analysis can also be used to construct symmetry functions in a systematic way. For instance, to find the η values for a set of G2 functions, first a function with $\eta = 0$ is constructed, which has the largest possible spatial extension. For the next function, the η value is increased, that is, the Gaussian is contracted, until the correlation of the obtained function values to the values of the first function decreases below a certain threshold value, for example, 90 %. Then, the next function is added and η is further increased until the correlation to the other two functions is below this threshold and so forth. The set of G2 functions is then complete, if an η value is reached, for which the range of values of the new symmetry function approaches zero.

All these tests for identifying a suitable set of symmetry functions depend on the specific composition of the available reference data set. Consequently, as the data set is typically increased step by step during the construction of a potential, several subsequent adjustments and refinements of the symmetry function set will be required. Usually it is not possible to use a large set of symmetry functions initially, since the small reference sets in the early stages of the construction of the NNP often only cover a part of the relevant configuration space. Functions, which enable to distinguish structural feature not present in this set may thus not have a sufficiently wide range of values. Conversely, the small symmetry function sets used for limited initial data sets will not allow to distinguish data entering this set at a later stage. Consequently, an extension of the set of symmetry functions will be required while the reference set is increasing.

Preconditioning the NN

To obtain a NNP, which is able to represent a given reference set with small errors, the fitting capabilities of NNs have to be exploited in an efficient way. This can be facilitated by preconditioning the atomic NNs in two different ways: concerning the input symmetry function values as well as concerning the initial values of the weight parameters.

By construction, the individual symmetry functions can have very different ranges of values, and a symmetry function with very large absolute values will have a siginificant impact on the value of the nodes in the first hidden layer, while symmetry functions with small absolute values will only play a minor role. The importance of the different symmetry functions can be balanced by rescaling the range of values for each symmetry function to the same interval,[60] for example, $[-1, 1]$, by applying

$$G_i^{\text{scaled}} = \frac{2\left(G_i - G_{i,\min}\right)}{G_{i,\max} - G_{i,\min}} - 1, \tag{12}$$

where $G_{i,\min}$ is the smallest value of function G_i and $G_{i,\max}$ its largest value in the data set. It is often useful to center the range of values around 0, as this is the center of the nonlinear regions of most activation functions. Alternatively, this can also be achieved by shifting the center of mass of all symmetry function values to zero. In addition to rescaling the symmetry functions, it is also possible to normalize the linear combinations at each node in the NN by dividing the sum by the number of nodes in the previous layer. This procedure has the advantage that it is applicable to arbitrary hidden layers and does not only refer to the first hidden layer. Further, even if the symmetry functions have been scaled, still the linear combination of symmetry function values can result in large numbers depending on the size of the NN, which is unfavorable as the subsequently applied activation functions are not used in their nonlinear region but in their saturation region. Also this problem can be solved by normalizing the linear combination before applying the activation functions.

Apart from preconditioning the symmetry function values, also the initial choice of the weight parameters is very important. They can be chosen simply as random numbers, but a decision has to be made on the type of distribution as well as on the range of values. Further, there is a number of procedures that have been proposed in the literature to determine an optimum set of initial weight parameters, like the scheme of Nguyen and Widrow.[106]

Finally, the initial errors of the training points can also be strongly reduced by preconditioning the weights. This primarily concerns the standard deviation and the center of mass of the output energies. The center of mass of the initially random energy outputs of the NNP can be aligned with the center of the target energies by adjusting the bias weight of the output node, which corresponds to a simple shift of the average output energy. The standard deviation of the NN output energies and of the energies in the training set can be matched by modifying the connecting weights between the last hidden layer and the output layer. By this simple procedure, the average initial errors of the NNP can often be reduced by one or two orders of magnitude.

Symmetry Functions and Forces

The availability of forces is crucial for many applications of NNPs like MD simulations and geometry optimizations. Since NNPs have well-defined functional forms, analytic derivatives are readily available by applying the chain rule to take into account the initial transformation of the atomic Cartesian coordinates onto the many-body symmetry functions. In general, the force with respect to some atomic coordinate α is the sum of the short range force $F_{\alpha,\text{s}}$ and the electrostatic force $F_{\alpha,\text{elec}}$,

$$F_\alpha = F_{\alpha,\text{s}} + F_{\alpha,\text{elec}} = -\frac{\partial E_\text{s}}{\partial \alpha} - \frac{\partial E_\text{elec}}{\partial \alpha}. \tag{13}$$

The short-range component of the force is then given by

$$\begin{aligned} F_{\alpha,\text{s}} &= -\frac{\partial E_\text{s}}{\partial \alpha} = -\sum_{j=1}^{N_\text{atom}} \frac{\partial E_j}{\partial \alpha} \\ &= -\sum_{j=1}^{N_\text{atom}} \sum_{\mu=1}^{N_{\text{sym},j}} \frac{\partial E_j}{\partial G_{j\mu}} \cdot \frac{\partial G_{j\mu}}{\partial \alpha} \quad , \end{aligned} \tag{14}$$

where $N_{\text{sym},j}$ is the number of symmetry functions of atom j. The first term in the product is given by the architecture of

the NN and contains also the weight parameters, while the second term depends on the definition of the symmetry functions.

The electrostatic force contribution can be calculated by

$$F_{\alpha,\mathrm{elec}} = -\frac{\partial}{\partial\alpha}\frac{1}{2}\sum_{i=1}^{N_{\mathrm{atom}}}\sum_{j=1,j\neq i}^{N_{\mathrm{atom}}}\frac{Q_iQ_j}{R_{ij}} = -\sum_{i=1}^{N_{\mathrm{atom}}}\sum_{j=1,j\neq i}^{N_{\mathrm{atom}}}\frac{1}{2R_{ij}^2}\left[\frac{\partial Q_i}{\partial\alpha}Q_jR_{ij}+Q_i\frac{\partial Q_j}{\partial\alpha}R_{ij}-Q_iQ_j\frac{\partial R_{ij}}{\partial\alpha}\right]. \quad (15)$$

Here, it is important to note that in addition to the usual derivative of Coulombs law that is used in many force fields using fixed atomic charges there are two additional terms taking into account the environment-dependence of the atomic charges Q_i and Q_j. After some rearrangements, regrouping of indices and considering the transformation of the Cartesian coordinates to symmetry functions we finally obtain

$$F_{\alpha,\mathrm{elec}} = \sum_{j=1}^{N_{\mathrm{atom}}}\sum_{i=1,i\neq j}^{N_{\mathrm{atom}}}\frac{Q_i}{R_{ij}}\cdot\left[\frac{1}{2}\frac{Q_j}{R_{ij}}\frac{\partial R_{ij}}{\partial\alpha}-\sum_{k=1}^{N_{\mathrm{sym},j}}\frac{\partial Q_j}{\partial G_{jk}}\frac{\partial G_{jk}}{\partial\alpha}\right]. \quad (16)$$

In principle, also higher derivatives can be calculated, but due to the complex nested functional form of the NN this can become rather complicated. Also other physical properties including gradients like the stress tensor of solids are accessible.[73]

It should be mentioned that the force acting on an atom depends on the positions of the atoms being as far as $2\cdot R_c$ away. At first glance, this seems odd, because the atomic energy contribution is determined only by the atoms inside the cutoff sphere. Still, this is fully consistent with the NN total energy expression, as the force with respect to a coordinate α of an atom i is the derivative of the energies E_j of all atoms j inside the cutoff sphere of i. In turn, the E_j depend on the positions of all atoms in their respective environments, and these atoms can be up to $2\cdot R_c$ away from atom i. As the positions of all atoms in the environment of j determine E_j, even an atom k being outside the cutoff sphere of atom i affects the contribution of atom i to E_j. Still, in practice this large effective range of the forces is not very important, since the effective physical range of the atomic interactions is usually much shorter than twice $R_c \approx 12-20$ Å.

Training NNs

Selecting the training data

In contrast to conventional physical potentials NNs have a very flexible functional form without a physical origin. Although this is the reason for the high accuracy that can be obtained in fitting the reference set, it can also give rise to large errors if NNPs are used to predict the energies for structures, which are very different from the configurations included in the training set.

There are two situations, when this can happen. First, NNs have very limited extrapolation capabilities, that is, they usually fail outside the range of input values spanned by the training data. This concerns even very basic properties of the PES like the strong repulsion of atoms at very short distances, which needs to be learned from example structures in the data set. It is often found in early stages of the NNP construction that structures collapse and adopt unphysically short interatomic distances or "explode." In general, these extrapolation cases are easy to identify. This is done by comparing the symmetry function vector of each atom in the structure with the minimum and maximum values of each symmetry function in the training data for the respective element. If a coordinate value is outside the range of validity of the potential, a warning can be issued and the simulation can be stopped. Although this should not happen in the final application of the completed NNP, this warning turns out to be very useful during the construction of the NNP as it is possible to search systematically for such extrapolating structures to extend and improve the reference set.

The other situation is more difficult to identify. Here, the energy is requested for an atomic configuration, which is within the range of validity of the symmetry functions, but still the structure is located in a part of the configuration space that is not well represented in the training set. Uncontrolled errors can be present for such NN energies and forces. In case they are still in the expected order of magnitude, this is very difficult to detect. The reason for these errors is the high flexiblity of NNs, which can cause strong artifacts in between sparsely distributed training points. This is the well-known "overfitting" problem, which is essentially a much better representation of the points in the training set than of the structures in between.

If the reference data set covers all parts of configuration space but is just too sparse, there is a simple recipe to detect overfitting, which is called the "early stopping" method. Here, the available reference data set is split into the training set, which is used for the optimization of the weight parameters, and an independent test or validation set, whose error is monitored during the fit but which is not used in the weight optimization. If the errors of the training and the test set are similar, the NNP has good generalization properties and it is applicable also to structures not included in the training set. If, on the other hand, the error of the test data is significantly larger than the error of the training set, overfitting is present and more training data is required. The typical evolution of a fit exhibiting overfitting is shown in Figure 6. In the first iterations the errors of both the training and the test set decrease, because the NN learns the overall shape of the PES. Then, the error of the test set reaches a minimum and starts to increase again slowly. At this stage, the NN is learning the fine details of the training points at the expense of a larger error of the test set, which is not visible to the NN optimization algorithm. Therefore, the best set of weight parameters corresponds to the epoch with the smallest test set error.

Unfortunately, the early stopping method is not generally applicable and is particularly problematic in case of high-dimensional PESs. Problems can arise, if there are parts of the configuration space in which no reference data are present at

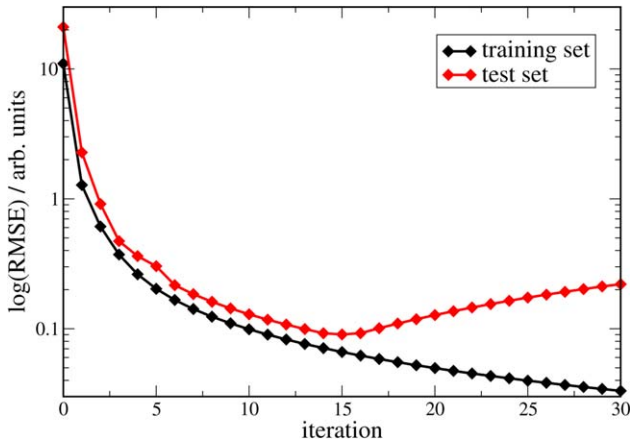

Figure 6. Illustration of the early stopping method. Initially, the errors of the training and the test set decrease as the NN learns the overall shape of the PES. Then, the test error reaches a minimum and starts to increase slowly, which labels the onset of overfitting (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

all, since in this case failures of the NNP would not be detected by a high error of the test set. Still, there is a possibility to check and improve the NNP also in this situation without the need to carry out costly electronic structure calculations for many structures just to validate the potential, which is anyway impractical in a systematic way for high-dimensional PESs. This validation is done by first fitting several NNPs to the same training set. These NNPs must have different functional forms, which can be achieved, for example, using atomic NNs with different architectures. Then, several NNPs of comparable quality are selected, which are apparently all equally applicable and accurate. Then a large number of structures is generated using one of these NNPs, for example, by running NN-driven MD simulations. Afterwards, the obtained trajectory is recalculated using the other NNPs and the energies along the trajectory are compared as shown in Figure 7. In panel (a), only a few structures representing the target function are used in the training and both NNPs agree well with the this function shown as black line close to these points. Far away from these points, however, the energy predictions of both NNPs are more or less random. This can be detected by comparing the predictions of several NNs. If there is a significant deviation between the predictions of the available NNPs for a given structure, then this structure is too far away from the training points, an electronic structure calculation should be carried out and it should be included in the training set to refine the NNP. The result is shown schematically in panel (b) of Figure 7. Here, the number of training points is increased while the region in which both NNPs deviate from each other is strongly reduced. After further improvements both NNPs agree well with the target function in panel (c) over the whole range. This multiple-fit validation should be carried out for the same simulation type and under the conditions of the intended applications of the NNP to ensure that the relevant part of configuration space is validated. As long as structures are found for which different fits having similar errors for the training set predict different energies or forces for some structures, the potential needs to be improved by adding further training structures to refine the values of the weight parameters. As soon as no problematic structures are found anymore, the NNP is reliable and the potential is ready for use. Still, ideally the final simulations should be carried out independently by different NNPs for validation.

Typically, the resulting reference data sets for the construction of NNPs are very large, starting from about 1000 structures to get an approximate PES for low-dimensional systems containing only a few atoms, up to several tens of thousands of structures for high-dimensional condensed systems. This poses some restrictions on the possible choice of the reference electronic structure method. To date, in most cases DFT has been employed, in particular for large systems, but for small molecules also a variety of higher-level wave function-based methods has been used. The reference method is very important for the scientific problem to be solved, as NNPs cannot provide better results than the underlying electronic structure method.

Concerning the composition of the training sets for high-dimensional NNPs, it is possible to combine different types of structures including periodic and nonperiodic systems and configurations with different numbers of atoms. Due to the reduced computational costs it is often helpful to start the NNP development with a set of structures for very small systems like thermally distorted crystal unit cells to get a first rough estimate for basic features of the PES like prefered interatomic distances and lattice constants. Then, as soon as NN-based simulations of these small systems work reliably, the size of the system can be increased step by step until the limit of what can be addressed by the reference electronic structure calculations has been reached. Still, often it is required to validate the potential also for systems that are much larger. In this case the effective reduction of dimensionality by the symmetry function cutoff can be exploited by cutting clusters centered at atoms that are not reliably represented by the NNP. These clusters, which can be identified, for example, by the multiple NNP method and investigating the forces as local probes of the PES, can then be recalculated by the reference method to include the environments of these atoms in the training set. In principle, it is sufficient to include the configurations that can be realized within the cutoff spheres in the training set to obtain a NNP that can be applied to systems of arbitrary size. Finally, electronic structure calculations for clusters can also be employed to directly validate the NN PES for very large systems using the forces. If the clusters are sufficiently large, which can be checked in convergence tests, the forces acting on the central atoms obtained in electronic structure calculations should be very similar to the NN forces for the full system, and this has indeed been found for several systems.[76,82]

In summary, the determination of the reference set can be done in a self-consistent way according to the following procedure:

1. Select an electronic structure method, which is suffiently fast to carry out the required number of reference

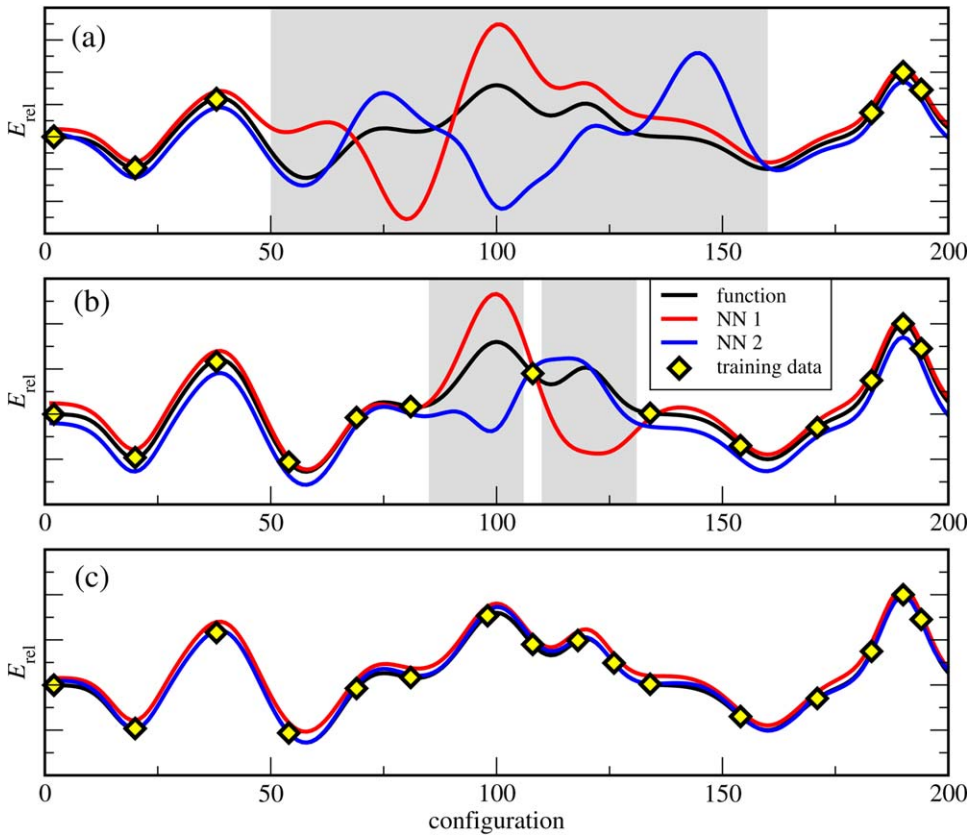

Figure 7. Illustration of the systematic improvement of the training set using the multiple-fit procedure. In panel (a), only a few training point of the target function (black line) are available and in between these points different NNs can predict very different energies E_{rel}. As more training points are added [panel (b)], the representation of the energy function becomes more reliable until a good fit is obtained over the full range in panel (c) (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

calculations and which is accurate enough to describe the physical properties of interest reliably.
2. Define a first set of structures and determine the reference energies and forces.
3. Construct a first preliminary NNP.
4. Carry out simulations using this NNP to find structures, which give rise to extrapolation warnings or unphysical geometries.
5. Determine the electronic structure energies and forces of these structures, include them in the training set and improve the NNP.
6. Improve the NNP systematically and self-consistently by running NNP-based simulations to find missing structures using the multiple-NNP method. Carry out electronic structure calculations for these structures, refine the fit and start again running extended simulations at the conditions of the intended applications.
7. As soon as no further problematic structures can be identified, the NNP is ready for use.

There is some flexibility in the choice of the first set of structures. In a "puristic" approach it is possible start with random structures to obtain an unbiased initial data set. This is more demanding as random structures typically have a high energy and will not be relevant in the final converged data set. Still, starting with such structures it is possible to improve the potential step by step until the correct structures are found. We have tested this for boron. Finally, after several iterative improvements employing metadynamics simulations,[107] we have obtained a potential suitable to identify the correct crystal structure of α-boron (Morawietz and Behler, unpublished). In a more "pragmatic" approach, which uses some knowledge about the system, the number of electronic structure calculations can be reduced by starting from reasonable atomic configurations. Further, also other types of potentials like classical force fields could be used to run the first simulations. In the later stages of the iterative refinement of the potential, it is still crucial to run simulations using the NNP itself to find problematic configurations like "holes" in the PES.

The large number of training points is one of the main remaining challenges of constructing NNPs. Still, there is some guidance how an excessive growth of the training set can be avoided. First of all, only the part of configuration space, which is accessible in MD simulations is relevant and needs to be mapped. Consequently, calculating the data on a regular grid is not required and is unfeasible anyway in high-dimensional systems as the number of points grows exponentially with the number of dimensions. Also using a metric to determine positions of candidate structures is not a promising approach as the "distance" between structures depends on the choice of

coordinates and can be very different in real space and in the symmetry function space. Further, this approach is sensitive to the preprocessing steps of the symmetry functions like scaling and centering. In general, as much information as possible should be extracted from each electronic structure calculation. Although there is just one total energy value, there are $3 \cdot N_{\text{atom}}$ force components containing valuable information about the local topology of the PES. This can be used to reduce the training set size significantly. It has also been proposed to employ the symmetry, for example, of single crystal surfaces,[49] but this is possible only in special applications as in general there is no symmetry left in MD simulations at finite temperatures.

Determination of the Weight Parameters

Once a set of reference data from electronic structure calculations is available, the central step in constructing a NNP is the determination of the numerical values of the weight parameters, which enable to reproduce these data as accurately as possible. In "supervised learning," which is the standard procedure in the training of NNPs, the output of the NN for each structure is compared to the known "true" answer from the reference calculations. In general, this corresponds to a minimization of an error function Γ, which is given as the sum of squared errors of the N_{struct} individual members of the training set,

$$\Gamma = \frac{1}{N_{\text{struct}}} \sum_{i=1}^{N_{\text{struct}}} \left(E_{\text{NN}}^i - E_{\text{Ref}}^i\right)^2. \tag{17}$$

If also the forces are used for the weight optimization, a modified error function,

$$\Gamma = \frac{1}{N_{\text{struct}}} \sum_{i=1}^{N_{\text{struct}}} \left[(E_{\text{NN}}^i - E_{\text{Ref}}^i)^2 + \frac{\beta}{3N_{\text{atom}}^i} \sum_{j=1}^{3N_{\text{atom}}^i} (F_{j\text{NN}}^i - F_{j\text{Ref}}^i)^2 \right] \quad , \tag{18}$$

can be used, which contains a loop over all *X*, *Y*, and *Z* force components of all atoms. The relative influence of the energies and forces can be balanced by the parameter β, as the number of force components is much larger than the number of energies. The value of β can be set to the inverse of the number of force components for each structure to take into account structures with different numbers of atoms, or to a predefined fixed value. Alternatively, we have found it useful to first update the weights using the error of the energy, followed by one update per force component. After each of these force updates the energy update is repeated to stress the importance of a correct representation of the energy.

The optimization of the weight parameters is done iteratively, and one iteration, which is often called "epoch" in the context of NNs, corresponds to a cycle through the full data in the training set. In "batch learning," which is also called offline learning, the weights are updated only once per iteration, while in "online learning" there is one weight update after the presentation of each piece of information, like energies or force components. Although online learning is computationally more demanding, in this procedure the fit is less likely to get trapped in local minima. This is an important advantage since there are typically several thousand parameters, and there is no hope to find the global minimum in this very high-dimensional optimization space. Still, usually many sufficiently accurate local minima are found, which provide a reliable description of the PES.

A wide range of optimization algorithms can be used to determine the weight parameters, and the standard procedure is to use gradient-based approaches. The most basic method is called "backpropagation" in the NN community,[108] which refers to the recursive calculation of the derivatives of the error function with respect to the weights starting from the output layer and proceeding to the input layer in the inverse order of the calculation of the NN output. In essence, the backpropagation algorithm corresponds to a steepest descent optimization of each weight *w*, which can be a connecting weight *a* or a bias weight *b*. For an iteration $t + 1$, the updated weights are obtained according to

$$w(t+1) = w(t) - \eta \cdot \frac{\partial \Gamma}{\partial w(t)} \quad . \tag{19}$$

Here, η is the learning rate, which can also be adapted during the progress of the fit. More advanced optimization algorithms that are frequently used for the determination of NN weights are the Levenberg-Marquardt algorithm[109] and in particular the global extended Kalman filter,[91,110–112] which we use for the development of high-dimensional NNPs. A discussion of the details of these optimization algorithms is beyond the scope of this review.

Regardless of the specific choice of optimization algorithm, for each energy the derivatives of the error function with respect to the connecting weights and the bias weights need to be determined for each atomic energy, as the reference energy is a constant and thus independent of the weights. Similar derivatives also need to be calculated for each force component. This has to be repeated after each weight update since the derivatives with respect to the weights depend also on their numerical values.

During the optimization, the quality of the fit is measured by determining the root mean squared error (RMSE) of the energies and forces in the training and the test set in each iteration using

$$\text{RMSE}(E) = \sqrt{\frac{1}{N_{\text{struct}}} \sum_{i=1}^{N_{\text{struct}}} \left(E_{\text{NN}}^i - E_{\text{Ref}}^i\right)^2} \tag{20}$$

and

$$\text{RMSE}(F) = \sqrt{\frac{1}{N_{\text{struct}}} \sum_{i=1}^{N_{\text{struct}}} \sum_{j=1}^{3N_{\text{atom}}^i} \frac{1}{3N_{\text{atom}}^i} \left(F_{j,\text{NN}}^i - F_{j,\text{Ref}}^i\right)^2}. \tag{21}$$

In particular, in case of large data sets including very different systems like cluster and bulk data, a global RMSE is not very helpful in understanding how accurate a fit will be in

which situation. For a more detailed analysis, the data set should be split in meaningful groups, like subsets of structures with the same number of atoms, and the error of each group should be determined for a given fit. To obtain a meaningful measure of the accuracy of the fit, the RMSE should also be normalized per atom to enable a comparison of different types of structures. Additionally, also points with particularly large errors should be identified and analyzed to understand the reason why they are not well-represented. A possible reason could be for instance an insufficient set of symmetry functions or problems in the underlying reference data. If specific parts of the configuration space, like transition states and local minima are of special interest, it is also possible to increase their impact on the fit,[28,49] but in many situations this distinction of the PES in more and less important parts is not straightforward.

Apart from the selection of the optimization algorithm, a number of additional choices have to be made to construct a specific NNP. First of all, the architecture of the atomic NNs has to be determined. This is still an empirical component of the NNP development, since the most efficient way to identify a suitable number of hidden layers and nodes per layer is simply to carry out a number of fits and to select the one with the lowest errors of the energies and forces in the test set. In general, the architecture of the NN determines its flexibility. Consequently, if the NN is too small, some features of the PES may not be resolved as shown in Figure 8a. This is visible in the RMSEs as a notable error in the training and the test set. If the flexbility of the NN is increased by using more nodes and possibly an additional hidden layer, the representation of the PES will improve, as demonstrated in Figure 8b, resulting in lower errors of the training and the test data. If the NN is too large, it becomes too flexible and overfitting can occur, which is illustrated in Figure 8c and can be identified by an increased error of the test set data compared to the error of the training data. The tendency for overfitting can be reduced significantly, if also forces are used for the training, because a correct representation of the gradient of the PES close to the training points will improve the description of neighboring structures.

Other choices in the fitting process can have a strong influence on the CPU time requirements. In particular, if also forces are used for the determination of the weight parameters, very large data sets often containing millions of pieces of information have to be processed in each iteration. This comprises first the determination of the error of each energy and force component, followed by the calculation of the derivative of this error with respect to each weight parameter and finally the update of the weights, which, for example, in case of the Kalman filter involves demanding matrix operations. Consequently, it should be avoided to perform unnecessary weight updates, for example, if an energy or force is already well represented. This can be done be introducing error thresholds and only if the error of an energy or force is above this threshold a weight update will be carried out. This threshold is usually coupled to the RMSE of the present iteration and, therefore, decreases along with the overall error of the NNP.

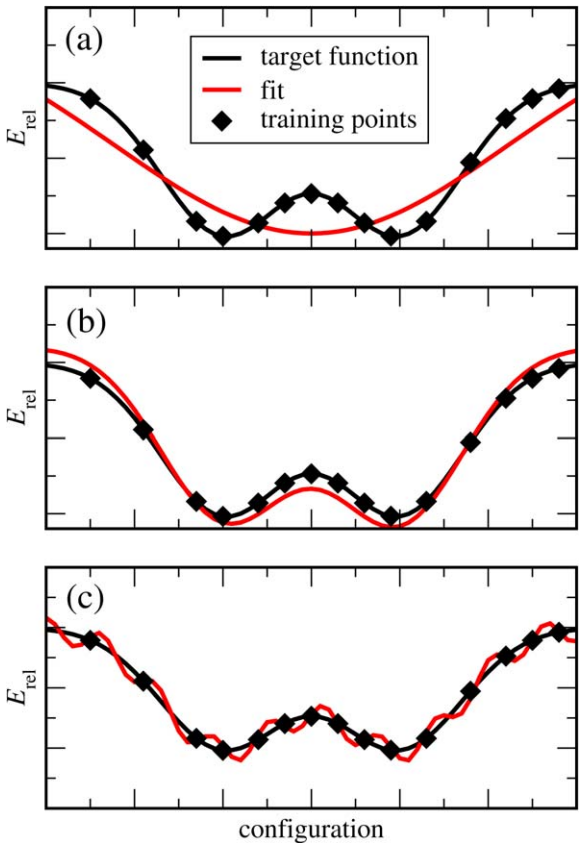

Figure 8. Influence of the flexibility of the NN on the fit quality. In panel (a), a very small NN is used, which is not flexible enough to resolve all features of the energy surface. In (b), the shape of the energy surface is well represented using a large NN, while in (c) overfitting is present as can be seen by the presence of artificial wiggles in between the training points (Figure reproduced from the "Jülich School on Computational Trends in Solvation and Transport in Liquids," Forschungszentrum Jülich 2015, under the CC-BY).

Another possibility to reduce the fitting effort is to use only a random subset of the forces in each iteration or to average the weight derivatives for several energies and/or forces and to perform a joint weight update after a certain number of weight derivatives has been accumulated.

Once the weights have been determined, the NNP is ready for use, and it is not relevant for the application how the weights have been obtained. Still, it is important to be aware that the "NNP" not only consists of the values of the weight parameters. Additionally, information about the architecture of the NN, the type of activation functions, the types and parameters of the symmetry functions, and possibly also scaling information for the preconditioning of the NN input nodes are required to apply a NNP in simulations. Ideally, the same NN software should be used for the simulations that has also been used for the construction of the NNP, since many subtle details of the implementation may complicate the transfer of this information from one code to another.

Discussion

Having reviewed the structure of high-dimensional NNPs, the functional form of the symmetry functions and the

determination of the training sets as well as of the weight parameters, some comments should be made concerning the general applicability and performance of high-dimensional NNPs. Like other ML methods, NNPs can be numerically very accurate and the typical RMSEs for the high-dimensional NNP method presented here are less than 10 meV/atom for potential-energies and between 50 and 200 meV/Bohr for the atomic forces. For energy differences, which are most important for the investigation of chemical and physical processes, the error can be even about one order of magnitude smaller. This has the consequence that replacing electronic structure methods like DFT in MD simulations by NNPs does not give rise to significant additional errors, which is crucial as substantial uncontrolled errors arising from the fit would not be acceptable. Conversely, the low RMSEs of NNPs pose stringent conditions on the convergence level of the training data, as any noise in the reference calculations would complicate the optimization of the weight parameters. As a rule of thumb, the convergence error of the energies and forces should be one order of magnitude smaller than the intended final RMSEs of the NNP. Another measure for the quality of NNPs is the comparison of observables related to the PES, like global and local minima[74,84] as well as vibrational frequencies,[83] to the underlying electronic structure method or even to experiment. Like for the energy, a very close agreement can be obtained.

Concerning the efficiency, NNPs are superior to any electronic structure method, and since there is one atomic NN per atom, the method scales linearly with the number of atoms. Further, it is straightforward to implement in parallel computer codes. On current desktop computers, depending on the complexity of the selected symmetry functions, the energies and forces of about 200 atoms can be calculated per second and per compute core. Still, this is substantially slower than the evaluation of basic classical force fields. NNPs, however, are no direct competitors of classical force fields due to their different applicability. NNPs are currently restricted to systems containing about three to four elements because of the rapidly increasing structural variety that would give rise to too many symmetry functions for systems consisting of more elements. This restriction is not present for force fields, which use only very low-dimensional additive terms. NNPs, conversely, are clearly superior to force fields concerning the numerical accuracy and their ability to work even in case of most complex geometries. Further, they are able to describe the making and breaking of bonds, which is a severe problem for most force fields. Like electronic structure methods, the input of NNPs consists of a description of the chemical elements and the atomic positions, but no classification of atoms according to functional groups or the definition of bonds is required. In contrast to common belief, also the number of fitting parameters in classical force fields can be very large taking into account the interaction between all the different atom types and bonds.

A clear drawback of NNPs is the nonphysical functional form, which does not allow for a physical interpretation. Still, this seems to be the price to be paid for obtaining potentials with a high numerical accuracy. Consequently, NNPs must be constructed and used with care. Since the functional form is not transparent to the user, NNPs should not be used as black box method but must be routinely tested using the methods described above.

There is still a number of open problems concerning the construction and application of high-dimensional NNPs. First, as the development of NNPs currently requires very large training sets, a major goal is to reduce the number of required reference data. One step in this direction is the use of forces apart from total energies only, which provide a lot of additional information about the local shape of the PES. Nevertheless, there is certainly room for a better and more selective choice of reference structures, although still all relevant parts of configuration space need to be mapped, which is posing a boundary on the minimum number of structures that need to be included. The limitation of NNPs to a few elements can be partially overcome by sacrificing the full generality of NNPs. If, for instance, it is known that certain bonds do not participate in chemical processes, the NNPs does not need to be able to describe the breaking and making of these bonds. Finally, also the fitting process itself can be very time consuming and most efficient online-learning algorithms like the Kalman filter cannot be parallelized well yet.

To date, NNPs have been applied successfully in different types of simulations like Monte Carlo,[59,60] MD,[28] metadynamics[74] and replica exchange MD simulations,[84] and it can be anticipated that the use of ML potentials and in particular of NNPs will rapidly increase in the near future.

Summary

In this tutorial review, a method to construct high-dimensional atomistic potentials employing artificial NNs has been presented and discussed. In contrast to conventional, low-dimensional NNPs, the potential-energy of the system is not represented by a single feed-forward NN. Instead, there is one separate NN for each atom in the system providing the contribution of that atom to the total energy. These atomic energy contributions depend on the local chemical environment defined by a cutoff radius resulting in atomic spheres containing up to about 100 atoms. The positions of these atoms with respect to the central atom are described by a vector of many-body symmetry functions serving as input for the atomic NNs. In addtion to this short-range energy also long-range electrostatic energies can be included based on environment-dependent charges represented by another set of atomic NNs.

The method is applicable to a wide range of systems and very low errors with respect to the underlying electronic structure data can be obtained. The evaluation of the NNP is several orders of magnitude faster than DFT even for systems of moderate size and scales linearly with the number of atoms. Still, due to the very flexible functional form of NNs the construction of the PES requires care and the obtained NNP must be thoroughly validated. The determination of the training set, which usually consists of tens of thousands of structures, is computationally demanding. An important limitation of the method is its current restriction to a few elements, although a large number of atoms of each element can be present, because of the rapidly increasing structural variety in the atomic environments,

which needs to be mapped by electronic structure calculations and has to be described by the symmetry functions.

Since NNPs are equally applicable to all types of bonding and even complex atomic environments, they are particularly useful for large-scale MD and Monte Carlo simulations of complex systems in materials science, at interfaces like the solid-liquid interface, and for studying solvation processes.

Acknowledgments

The author thanks the Forschungszentrum Jülich for the permission to publish this review using his lecture notes of the "Jülich School on Computational Trends in Solvation and Transport in Liquids" taking place March 23–27, 2015, at the Jülich Supercomputing Center. This work was supported by the Cluster of Excellence RESOLV (EXC 1069) funded by the Deutsche Forschungsgemeinschaft as well as by the Emmy Noether program (BE3264/3-1) and a Heisenberg fellowship (BE3264/6-1) of the DFG.

Keywords: neural network potentials · molecular dynamics

How to cite this article: J., Behler *Int. J. Quantum Chem.* **2015**, *115*, 1032–1050. DOI: 10.1002/qua.24890

[1] M. Born, R. Oppenheimer, *Ann. Phys.* **1927**, *389*, 457.
[2] R. Car, M. Parrinello, *Phys. Rev. Lett.* **1985**, *55*, 2471.
[3] D. Marx, J. Hutter, Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods; Cambridge University Press: Cambridge, **2009**.
[4] R. G. Parr, W. Yang, Density Functional Theory of Atoms and Molecules; Oxford University Press: Oxford, **1989**.
[5] W. Koch, M. C. Holthausen, A Chemist's Guide to Density Functional Theory, 2nd ed.; Wiley-VCH, **2001**.
[6] N. L. Allinger, Y. H. Yuh, J.-H. Lii, *J. Am. Chem. Soc.* **1989**, *111*, 8551.
[7] W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, Jr., K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell, P. A. Kollman, *J. Am. Chem. Soc.* **1995**, *117*, 5179.
[8] S. L. Mayo, B. D. Olafson, W. A. Goddard, III, *J. Phys. Chem.* **1990**, *94*, 8897.
[9] A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard, III, W. M. Skiff, *J. Am. Chem. Soc.* **1992**, *114*, 10024.
[10] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, M. Karplus, *J. Comput. Chem.* **1983**, *4*, 187.
[11] F. H. Stillinger, T. A. Weber, *Phys. Rev. B* **1985**, *31*, 5262.
[12] J. Tersoff, *Phys. Rev. Lett.* **1986**, *56*, 632.
[13] M. W. Finnis, *Prog. Mater. Sci.* **2007**, *52*, 133.
[14] A. C. T. van Duin, S. Dasgupta, F. Lorant, W. A. Goddard, III, *J. Phys. Chem. A* **2001**, *105*, 9396.
[15] C. A. Coulson, *Proc. R. Soc. Lond.* **1939**, *A169*, 413.
[16] A. Brown, B. J. Braams, K. Christoffel, Z. Jin, J. M. Bowman, *J. Chem. Phys.* **2003**, *119*, 8790.
[17] J. M. Bowman, G. Czakó, B. Fu, *Phys. Chem. Chem. Phys.* **2011**, *13*, 8094.
[18] J. Ischtwan, M. A. Collins, *J. Chem. Phys.* **1994**, *100*, 8080.
[19] C. Crespos, M. A. Collins, E. Pijper, G. J. Kroes, *Chem. Phys. Lett.* **2003**, *376*, 566.
[20] A. P. Bartók, M. C. Payne, R. Kondor, G. Csányi, *Phys. Rev. Lett.* **2010**, *104*, 136403.
[21] A. P. Bartók, M. J. Gillan, F. R. Manby, G. Csányi, *Phys. Rev. B* **2013**, *88*, 054104.
[22] M. Rupp, A. Tkatchenko, K.-R. Müller, O. A. von Lilienfeld, *Phys. Rev. Lett.* **2012**, *108*, 058301.
[23] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, K.-R. Müller, *J. Chem. Theory Comp.* **2013**, *9*, 3404.
[24] M. J. L. Mills, P. L. A. Popelier, *Theor. Chem. Acc.* **2012**, *131*, 1137.
[25] T. Ishida, G. C. Schatz, *Chem. Phys. Lett.* **1999**, *314*, 369.
[26] R. Dawes, D. L. Thompson, Y. Guo, A. F. Wagner, M. Minkoff, *J. Chem. Phys.* **2007**, *126*, 184108.
[27] T. B. Blank, S. D. Brown, A. W. Calhoun, D. J. Doren, *J. Chem. Phys.* **1995**, *103*, 4129.
[28] S. Lorenz, A. Groß, M. Scheffler, *Chem. Phys. Lett.* **2004**, *395*, 210.
[29] A. Vitek, M. Stacho, P. Kromer, V. Snasel, In: 5th IEEE International Conference on Intelligent Networking and Collaborative Systems (INCoS), **2013**, pp. 121–126.
[30] C. M. Handley, J. Behler, *Eur. Phys. J. B* **2014**, *87*, 152.
[31] C. M. Bishop, Neural Networks for Pattern Recognition; Oxford University Press: Oxford, **1996**.
[32] S. Haykin, Neural Networks and Learning Machines, 3rd International ed.; Prentice Hall: New York, **2009**.
[33] W. McCulloch, W. Pitts, *Bull. Math. Biophys.* **1943**, *5*, 115.
[34] J. Gasteiger, J. Zupan, *Angew. Chem. Int. Ed.* **1993**, *32*, 503.
[35] T. Kohonen, *Neural Networks* **1988**, *1*, 3.
[36] G. Cybenko, *Math. Control Signals Syst.* **1989**, *2*, 303.
[37] K. Hornik, M. Stinchcombe, H. White, *Neural Networks* **1989**, *2*, 359.
[38] F. Scarselli, A. V. Tsoi, *Neural Networks* **1998**, *11*, 15.
[39] J.-G. Attali, G. Pages, *Neural Networks* **1997**, *10*, 1069.
[40] K. Hornik, *Neural Networks* **1991**, *4*, 251.
[41] J. Behler, *J. Phys. Condens. Matter* **2014**, *26*, 183001.
[42] F. V. Prudente, J. J. Soares Neto, *Chem. Phys. Lett.* **1998**, *287*, 585.
[43] F. V. Prudente, P. H. Acioli, J. J. Soares Neto, *J. Chem. Phys.* **1998**, *109*, 8801.
[44] K. T. No, B. H. Chang, S. Y. Kim, M. S. Jhon, H. A. Scheraga, *Chem. Phys. Lett.* **1997**, *271*, 152.
[45] H. M. Le, S. Huynh, L. M. Raff, *J. Chem. Phys.* **2009**, *131*, 014107.
[46] H. M. Lee, L. M. Raff, *J. Chem. Phys.* **2008**, *128*, 194310.
[47] J. Ludwig, D. G. Vlachos, *J. Chem. Phys.* **2007**, *127*, 154716.
[48] J. Behler, S. Lorenz, K. Reuter, *J. Chem. Phys.* **2007**, *127*, 014705.
[49] S. Lorenz, M. Scheffler, A. Groß, *Phys. Rev. B* **2006**, *73*, 115431.
[50] J. Behler, B. Delley, S. Lorenz, K. Reuter, M. Scheffler, *Phys. Rev. Lett.* **2005**, *94*, 36104.
[51] J. Behler, K. Reuter, M. Scheffler, *Phys. Rev. B* **2008**, *77*, 115421.
[52] C. Carbogno, J. Behler, A. Groß, K. Reuter, *Phys. Rev. Lett.* **2008**, *101*, 096104.
[53] C. Carbogno, J. Behler, K. Reuter, A. Groß, *Phys. Rev. B* **2010**, *81*, 035410.
[54] I. Goikoetxea, J. Beltrán, J. Meyer, J. I. Juaristi, M. Alducin, K. Reuter, *New J. Phys.* **2012**, *14*, 013050.
[55] I. Goikoetxea, J. Meyer, J. I. Juaristi, M. Alducin, K. Reuter, *Phys. Rev. Lett.* **2014**, *112*, 156101.
[56] D. A. R. S. Latino, R. P. S. Fartaria, F. F. M. Freitas, J. Aires-de-Sousa, F. M. S. S. Fernandes, *J. Electroanal. Chem.* **2008**, *624*, 109.
[57] S. Manzhos, K. Yamashita, *Surf. Sci.* **2010**, *604*, 554.
[58] T. Liu, B. Fu, D. H. Zhang, *Sci. China Chem.* **2014**, *57*, 147.
[59] K.-W. Cho, K. T. No, H. A. Scheraga, *J. Mol. Struct.* **2002**, *641*, 77.
[60] H. Gassner, M. Probst, A. Lauenstein, K. Hermansson, *J. Phys. Chem. A* **1998**, *102*, 4596.
[61] C. M. Handley, P. L. A. Popelier, *J. Phys. Chem. A* **2010**, *114*, 3371.
[62] J. Behler, *Phys. Chem. Chem. Phys.* **2011**, *13*, 17930.
[63] S. Hobday, R. Smith, J. Belbruno, *Model. Simul. Mater. Sci. Eng.* **1999**, *7*, 397.
[64] S. Hobday, R. Smith, J. BelBruno, *Nucl. Instrum. Methods Phys. Res. B* **1999**, *153*, 247.
[65] A. Bholoa, S. D. Kenny, R. Smith, *Nucl. Instrum. Methods Phys. Res. B* **2007**, *255*, 1.
[66] E. Sanville, A. Bholoa, R. Smith, S. D. Kenny, *J. Phys. Condens. Matter* **2008**, *20*, 285219.
[67] S. Manzhos, T. Carrington, Jr., *J. Chem. Phys.* **2006**, *125*, 84109.
[68] S. Manzhos, T. Carrington, Jr., *J. Chem. Phys.* **2006**, *125*, 194105.
[69] S. Manzhos, T. Carrington, Jr., *J. Chem. Phys.* **2007**, *127*, 014103.
[70] S. Manzhos, T. Carrington, Jr., *J. Chem. Phys.* **2008**, *129*, 224104.
[71] S. Manzhos, K. Yamashita, T. Carrington, Jr., *Comp. Phys. Comm.* **2009**, *180*, 2002.
[72] J. Behler, M. Parrinello, *Phys. Rev. Lett.* **2007**, *98*, 146401.
[73] J. Behler, *J. Chem. Phys.* **2011**, *134*, 074106.

[74] J. Behler, R. Martoňák, D. Donadio, M. Parrinello, *Phys. Rev. Lett.* **2008**, *100*, 185501.
[75] J. Behler, R. Martoňák, D. Donadio, M. Parrinello, *Phys. Status Solidi B* **2008**, *245*, 2618.
[76] N. Artrith, J. Behler, *Phys. Rev. B* **2012**, *85*, 045439.
[77] R. Z. Khaliullin, H. Eshet, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **2010**, *81*, 100103.
[78] R. Z. Khaliullin, H. Eshet, T. D. Kühne, J. Behler, M. Parrinello, *Nat. Mater.* **2011**, *10*, 693.
[79] H. Eshet, R. Z. Khaliullin, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **2010**, *81*, 184107.
[80] H. Eshet, R. Z. Khaliullin, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. Lett.* **2012**, *108*, 115701.
[81] N. Artrith, T. Morawietz, J. Behler, *Phys. Rev. B* **2011**, *83*, 153101.
[82] N. Artrith, B. Hiller, J. Behler, *Phys. Status Solidi B* **2013**, *250*, 1191.
[83] T. Morawietz, V. Sharma, J. Behler, *J. Chem. Phys.* **2012**, *136*, 064103.
[84] T. Morawietz, J. Behler, *J. Phys. Chem. A* **2013**, *117*, 7356.
[85] T. Morawietz, J. Behler, *Z. Phys. Chem.* **2013**, *227*, 1559.
[86] S. Kondati Natarajan, T. Morawietz, J. Behler, *Phys. Chem. Chem. Phys.* (in press). DOI: 10.1039/C4CP04751F.
[87] G. C. Sosso, G. Miceli, S. Caravati, J. Behler, M. Bernasconi, *Phys. Rev. B* **2012**, *85*, 174103.
[88] G. C. Sosso, D. Donadio, S. Caravati, J. Behler, M. Bernasconi, *Phys. Rev. B* **2012**, *86*, 104301.
[89] G. C. Sosso, J. Behler, M. Bernasconi, *Phys. Status Solidi B* **2012**, *249*, 1880.
[90] G. C. Sosso, G. Miceli, S. Caravati, F. Giberti, J. Behler, M. Bernasconi, *J. Phys. Chem. Lett.* **2013**, *4*, 4241.
[91] J. B. Witkoskie, D. J. Doren, *J. Chem. Theory Comput.* **2005**, *1*, 14.
[92] A. Pukrittayakamee, M. Malshe, M. Hagan, L. M. Raff, R. Narulkar, S. Bukkapatnum, R. Komanduri, *J. Chem. Phys.* **2009**, *130*, 134101.
[93] C. Muñoz-Caro, A. Niño, *Comput. Chem.* **1998**, *22*, 355.
[94] P. P. Ewald, *Ann. Phys.* **1921**, *64*, 253.
[95] P. Salvador, I. Mayer, *J. Chem. Phys.* **2007**, *126*, 234113.
[96] I. Mayer, *Phys. Chem. Chem. Phys.* **2006**, *8*, 4630.
[97] F. L. Hirshfeld, *Theor. Chim. Acta* **1977**, *44*, 129.
[98] S. Houlding, S. Y. Liem, P. L. A. Popelier, *Int. J. Quantum Chem.* **2007**, *107*, 2817.
[99] M. G. Darley, C. M. Handley, P. L. A. Popelier, *J. Chem. Theor. Comput.* **2008**, *4*, 1435.
[100] S. Grimme, *WIREs Comput. Mol. Sci.* **2011**, *1*, 211.
[101] S. Grimme, J. Antony, S. Ehrlich, H. Krieg, *J. Chem. Phys.* **2010**, *132*, 154104.
[102] A. Tkatchenko, M. Scheffler, *Phys. Rev. Lett.* **2009**, *102*, 073005.
[103] B. J. Braams, J. M. Bowman, *Int. Rev. Phys. Chem.* **2009**, *28*, 577.
[104] P. Geiger, C. Dellago, *J. Chem. Phys.* **2013**, *139*, 164105.
[105] K. V. Jovan Jose, N. Artrith, J. Behler, *J. Chem. Phys.* **2012**, *136*, 194111.
[106] D. H. Nguyen, B. Widrow, *IEEE Control Syst. Mag.* **1990**, *3*, 18.
[107] A. Laio, M. Parrinello, *Proc. Nat. Acad. Sci. USA* **2002**, *99*, 12562.
[108] D. E. Rumelhart, G. E. Hinton, R. J. Williams, *Nature* **1986**, *323*, 533.
[109] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical Recipes - The Art of Scientific Computing; Cambridge University Press: Cambridge, **2007**.
[110] R. E. Kalman, *J. Basic Eng.* **1960**, *82*, 35.
[111] S. Haykin, Kalman Filtering and Neural Networks; Wiley: New York, **2001**.
[112] T. B. Blank, S. D. Brown, *J. Chemometrics* **1994**, *8*, 391.

Received: 29 December 2014
Revised: 8 February 2015
Accepted: 10 February 2015
Published online 6 March 2015