

Next generation interatomic potentials for condensed systems

Christopher Michael Handley and Jörg Behler^a

Lehrstuhl für Theoretische Chemie, Ruhr-Universität Bochum, 44780 Bochum, Germany

Received 30 January 2014

Published online 7 July 2014 – © EDP Sciences, Società Italiana di Fisica, Springer-Verlag 2014

Abstract. The computer simulation of condensed systems is a challenging task. While electronic structure methods like density-functional theory (DFT) usually provide a good compromise between accuracy and efficiency, they are computationally very demanding and thus applicable only to systems containing up to a few hundred atoms. Unfortunately, many interesting problems require simulations to be performed on much larger systems involving thousands of atoms or more. Consequently, more efficient methods are urgently needed, and a lot of effort has been spent on the development of a large variety of potentials enabling simulations with significantly extended time and length scales. Most commonly, these potentials are based on physically motivated functional forms and thus perform very well for the applications they have been designed for. On the other hand, they are often highly system-specific and thus cannot easily be transferred from one system to another. Moreover, their numerical accuracy is restricted by the intrinsic limitations of the imposed functional forms. In recent years, several novel types of potentials have emerged, which are not based on physical considerations. Instead, they aim to reproduce a set of reference electronic structure data as accurately as possible by using very general and flexible functional forms. In this review we will survey a number of these methods. While they differ in the choice of the employed mathematical functions, they all have in common that they provide high-quality potential-energy surfaces, while the efficiency is comparable to conventional empirical potentials. It has been demonstrated that in many cases these potentials now offer a very interesting new approach to study complex systems with hitherto unreached accuracy.

1 Introduction

For the computer simulation of condensed phase systems, there are two fundamental approaches to determine the required energies and forces. The first is the use of electronic structure techniques. With increasing computer resources, the applicability of these methods has been extended continuously in recent years, and many different methods are now available. In particular density-functional theory (DFT) [1] has become a standard method in *ab initio* molecular dynamics (MD) simulations [2,3] of condensed systems with many successful applications. Still, while accurate, these simulations require large amounts of CPU time, and in many cases detailed simulations are still unaffordable. The computational costs of electronic structure calculations can be notably reduced by introducing additional approximations. An example is the family of tight-binding methods [4–7], which allows for the study of much larger systems. Further coarse graining steps building on tight-binding can be made resulting, for instance, in the bond order potentials developed by Hammerschmidt et al. [8].

The other option is to use atomistic potentials, in which the electronic degrees of freedom are not taken into account explicitly. Instead, a direct functional relation between the atomic configuration and the potential-energy

is derived, which is typically based on physical considerations and simplifications. The validity of this procedure can be deduced from quantum mechanics. Employing the Born-Oppenheimer approximation, the Hamilton operator of the electronic Schrödinger equation is fully defined by specifying the atomic positions, the chemical elements, and the total charge of the system, which is typically neutral. Since the sum of the electrostatic repulsion energy between the nuclei and of the eigenvalue of the electronic Hamilton operator yields the potential-energy of a given atomic configuration, a direct functional relation between the structure and its energy must exist, which corresponds to the potential-energy surface (PES). If the PES would be known analytically, very efficient simulations could be carried out by avoiding demanding electronic structure calculations. Unfortunately, the development of atomistic potentials, or equivalently of PESs, is a very challenging task, since their multidimensional topology is governed by the full complexity of the underlying quantum mechanical many-electron problem, which is very difficult to describe by closed analytic forms in all but the most simple systems.

PESs for large-scale simulations can be constructed in many different ways. The typical approach in classical force field methods [9,10], which are frequently used in MD simulations of biochemical systems, is to construct the PES as a sum of simple low-dimensional terms

^a e-mail: joerg.behler@theochem.ruhr-uni-bochum.de

representing covalent bonds, angles, and dihedral angles, and to consider non-bonded electrostatic and dispersion interactions. By using a pre-defined bonding pattern and atom types based on functional groups, most classical force fields are unable to describe chemical reactions or significant atomic rearrangements. Consequently they cannot be used to study many problems in materials science involving, e.g., metals and semiconductors, or to address questions related to the making and breaking of bonds. Only a few methods employing the ideas of classical force fields, like the reactive force field ReaxFF [11] or empirical valence bond (EVB) potentials [12] are able to overcome these limitations to some extent.

For theoretical studies in the field of materials science many of the simplifications used in classical force fields have to be abandoned. Instead, atomistic potentials must be able to describe a close to continuous range of atomic configurations and chemical environments very accurately. Due to the large diversity of relevant structures, from different crystal structures via all kinds of local and extended defects to amorphous solids and glasses, the distinction between bonded and non-bonded atoms is usually not possible and suitable potentials must employ functions based on the atomic positions only.

The selection of an appropriate functional form is only the start of the creation of a PES. Typically, a number of parameters needs to be optimized in order to obtain an accurate representation of the atomic interactions. These parameters are determined by minimizing the errors of the energy, the forces, and in a few cases the Hessian for a number of example configurations, for which reference electronic structure calculations have been carried out. Alternatively, some potentials are also making use of available experimental data, and also a combination of experimental and theoretical data is possible. The identification of the optimum set of parameter values can be a laborious task, since the final potential must yield accurate results for a substantial number of different physical properties, like binding energies, equilibrium geometries, elastic constants, and vibrational (or phonon) frequencies.

Countless potentials have been published to date for all imaginable systems and applications. The Tersoff potential [13,14], which has been frequently used for the investigation of semiconductors. Brenner presented an extension of the method to allow for the simulation of hydrocarbons, the so-called REBO (reactive bond order) potential [15]. Here, the bond order is chosen from a selection of values, where the appropriate choice is driven by the nature of the local chemical environment of the bond. This extension of the work by Tersoff also included the addition of new interaction terms that were based upon the degree of conjugation of the bond.

Another method, which is very popular for studying metals and alloys, is the embedded atom method (EAM) by Pijper et al. [16] and Daw et al. [17]. EAM in its most basic form consists of two terms, a repulsive part, and an attractive part. The repulsive term is a pairwise potential that depends only on the interatomic distance be-

tween two atoms. The second term, the attractive part, is the embedding potential. It determines the energy change when bringing a free atom into a position within the material, and thus into the electron density of all other atoms. The energy of an atom at a given place must then be determined using the embedding function, which references the electron density at that point being determined from a superposition of atomic densities of the other atoms. An extension, which also takes angular variations of the electron density into account, is the modified embedded atom method (MEAM) [18], which improves the accuracy of the approach significantly.

Related to explicit electronic structure methods is the family of tight binding approaches [4,5,7]. Tight binding assumes that the electrons of the atoms are “tightly bound” – hence the name – and that the atoms and their associated electrons are mainly interacting with their close neighbors. Because of this assumption atomic-like orbitals form a natural basis set for the interacting atoms, and a linear combination of these orbitals is used to construct the many-body Hamiltonian. Rather than computing the integrals describing the interaction between these orbitals, the Hamiltonian is instead constructed from parametrized functions, where each function is determined for each atom pair combination, and found by fitting to quantum mechanical calculations. These functions often depend only on the parameters and the distance between atoms and the direction cosines which describe the relative orientation and alignment of these orbitals [19–21]. Solving the generalized eigenvalue equation for these matrices then returns the eigenvalues and eigenvectors providing further electronic properties of the system, such as the band structure or the crystal field splitting of *d*-orbitals. The Slater Koster integrals of tight binding [19] do have a parallel for molecular systems – Extended Huckel Theory [22]. Further molecular extensions are the Angular Overlap Model, and Ligand Field Molecular Mechanics, enabling the description of *d*-orbitals within classical force fields [23,24].

Countless other potentials have been published in recent decades for performing computer simulations of materials, and a comprehensive overview is beyond the scope of this review. All these approaches have in common that they are typically based on some physical approximations to reduce the complexity of the potentials to a tractable level. In most cases the resulting atomistic potentials are thus very efficient and they perform reasonably well for the intended applications, but it is very challenging to obtain high-quality energies and forces close to first-principles data. The limited numerical accuracy has its origin in the functional forms, which are often less flexible and enforce certain properties of the potential. They usually contain only a few fitting parameters and consequently they are too inflexible to capture all the fine details of complex energy landscapes.

In this review we will survey a variety of methods for constructing PESs, which aim to overcome the limitations of conventional potentials by choosing very generic functional forms that are not based on physical considerations

or approximations. Instead, the goal of these “mathematical potentials” is to reproduce a given set of reference data from electronic structure calculations as accurately as possible. Due to the purely mathematical nature of these potentials they are able to describe all types of bonding, from dispersion interactions via covalent bonds to metallic systems, with similar accuracy employing the same functional form. Many of these methods, which have emerged only in recent years in the literature in the context of PESs, are often also summarized under the keyword “machine learning” potentials. They can be constructed using a variety of different functional units like Gaussians, Taylor series expansions, sigmoid-like functions and many others, and often a superposition or some other form of combination of a large number of these functions is used to express complicated multidimensional PESs. Atomistic potentials of this type are still not widely distributed, but their development has seen enormous progress in recent years and they have now reached a level of maturity making them interesting candidates for solving problems in chemistry and condensed matter physics.

When constructing accurate atomistic potentials, a number of substantial challenges have to be met.

1. *Accuracy* – The energy and forces provided by the potential should be as close as possible to the data of reference electronic structure calculations. A mandatory condition to reach this goal is that the potential is sufficiently high-dimensional. Only then it will be able to capture the effects of polarization and charge transfer, and the consideration of many-body effects is also most important for describing metals. Further, the potential should be transferable, since a potential is only useful if it is applicable to structures, which have not been used for its construction.
2. *Efficiency* – Once constructed, the potential should be as fast as possible to evaluate to enable the extension of the time and length scales of atomistic simulations clearly beyond the realm of electronic structure methods.
3. *Generality* – The potential should have a universal and unbiased functional form, i.e., it should describe all types of atomic interactions with the same accuracy. Further, the functional form should enable the calculation of analytic derivatives to determine forces needed in molecular dynamics simulations.
4. *Reactivity* – The potential should not require the specification of fixed bonding patterns or a further discrimination of different atom types for a given element. Instead it should be able to describe the making and breaking of bonds and arbitrary structural changes, which is equivalent to depending solely on the positions and nuclear charges of the atoms.
5. *Automation* – The parametrization of the potential for a specific system should require a minimum of human effort.
6. *Costs* – The amount of demanding electronic structure data required to construct the potential should be as small as possible.

Apart from criterion three, all these points are equally important for “physical” and “mathematical” potentials, and meeting all these requirements to full satisfaction is very difficult to achieve for any type of potential currently available.

Some further aspects of the construction of potentials need to be discussed here, which are specific for purely mathematical potentials, while they usually do not represent serious problems for conventional potentials. They concern some invariances of the PES, that need to be incorporated in a proper way. First, the potential-energy of a system must be invariant with respect to rotation and translation of the system, since only the relative atomic positions are important for the energy and forces. Further, the energy of a system must not change upon interchange of the positions of any two atoms of the same chemical element. Including this permutation invariance is straightforward, e.g., in classical force fields, which express the energy as a sum of many individual low-dimensional terms, or in electronic structure methods, which incorporate this invariance by the diagonalization of the Hamiltonian. In case of high-order many-body atomistic potentials, which are required to reach an accuracy comparable to that of first-principles methods, including this permutation invariance is one of the main conceptual challenges. In the following sections, some methods, which have evolved in recent years as general-purpose potentials for a wide range of systems, are highlighted, and their properties with respect to the criteria mentioned above will be discussed.

The ultimate goal of such atomistic potentials is the simulation of large periodic and non-periodic systems, ranging from large molecules, bulk liquids, and solids to extended interfaces, for which electronic structure methods are computationally too demanding to use, or are not even tractable.

2 General-purpose atomistic potentials

2.1 Polynomial fitting

In 2003 Brown et al. introduced a method employing polynomials as basis functions to construct PESs of molecular systems [25]. The structural description of the system starts with a full set of interatomic distances, like $\{R_{12}, R_{13}, R_{14}, R_{23}, R_{24}, R_{34}\}$ for the example of a tetra-atomic molecule A_4 . The R_{ij} are then transformed to a set of Morse variables defined as:

$$Y_{ij} = \exp(-R_{ij}/\gamma), \quad (1)$$

with γ being a constant between 1.5 and 3 Bohr. Since distances as well as Morse variables represent internal coordinates, whose values do not change with rotation and translation of the molecule, this description ensures translational and rotational invariance of the potential. A full distance matrix is, however, not invariant with respect to a permutation of like atoms. Therefore, a straightforward

construction of the PES as a linear combination of polynomials [26],

$$E(Y_{12}, Y_{13}, Y_{14}, Y_{23}, Y_{24}, Y_{34}) = \sum_{a+b+c+d+e+f=0}^{n_{\max}} C_{a,b,c,d,e,f} \left[Y_{12}^a, Y_{13}^b, Y_{14}^c, Y_{23}^d, Y_{24}^e, Y_{34}^f \right], \quad (2)$$

with coefficients $C_{a,b,c,d,e,f}$, does not exhibit this invariance. Here, a , b , c , d , e , and f are non-negative integers with a sum between zero and the maximum order n_{\max} of the polynomials to be considered. Typically, the largest exponents used are five or six.

In order to obtain a permutationally invariant basis set, the monomials are symmetrized employing a symmetry operator S ,

$$E(Y_{12}, Y_{13}, Y_{14}, Y_{23}, Y_{24}, Y_{34}) = \sum_{a+b+c+d+e+f=0}^{n_{\max}} D_{a,b,c,d,e,f} S \left[Y_{12}^a, Y_{13}^b, Y_{14}^c, Y_{23}^d, Y_{24}^e, Y_{34}^f \right], \quad (3)$$

which yields the same basis functions irrespective of the order of chemically equivalent atoms. All coefficients $D_{a,b,c,d,e,f}$ of the symmetrized permutationally invariant polynomials are fitted simultaneously using least-squares fitting procedures to a database of high-level ab initio calculations, which contains typically several tens of thousands of data points. The specific form of the symmetrized terms, whose number can be substantial, depends on the chemical composition of the molecule, and to date the method has been applied to systems containing up to 10 atoms. More details about the symmetrization procedure, for which efficient methods are available [26], and the resulting terms can be found elsewhere [27].

Although the complexity of the method currently requires a truncation of the order of the polynomials at comparably low order and restricts the formal dimensionality to that of small molecules, the resulting potentials can still be applied to large systems like bulk water using two- and three-body terms only [28]. This is possible since in molecular systems high-order many-body interactions are usually not essential. Further examples for successful applications of the method are various charged and neutral water clusters [26,29,30], the vinyl radical [31] and many others. Thus, while the construction of PESs using polynomials is a powerful technique for the investigation of dynamics, reactions, dissociation, and conformational changes for small molecules or systems composed of these, there has been no work towards implementing the technique with respect to the creation of accurate potentials that can be used to model systems like metals and semiconductors.

2.2 Gaussian processes regression

Gaussian process regression (GPR) [32], also known as Kriging, is a method that models observables as a realization of a underlying statistical probabilistic model –

the probability density function. The basic concept is that training examples with correlated observables, i.e., energies, have also a close correlation in their variables, i.e., the coordinates describing the atomic configuration. The assumption then is that the function, which models the non-linear relationship between the variables and the observables, the PES, is smoothly varying. The underlying statistical model, the probability density function determines the probabilities that for a given set of inputs a particular output is observed.

Since the output is known for a number of configurations from reference electronic structure calculations, the reverse question can be addressed: Can the probability density function be found that best models the PES for a given number of structures and their associated energies and takes into account the reliability of predictions in the regions near to the known reference points?

All reference data for the GPR model come from a deterministic function which operates on a set of input coordinates, where reference point i has coordinates $\mathbf{x}^i = (x_1^i, \dots, x_{N_{\text{dim}}}^i)$, and N_{dim} is the number of dimensions or degrees of freedom of the system.

$$E(\mathbf{x}^i) = z(\mathbf{x}^i) + \sum_h \beta_h f_h(\mathbf{x}^i). \quad (4)$$

The f_h are some linear or non-linear functions of \mathbf{x} , and β are coefficients, where h runs over all functions that define the PES. It then follows that, because the reference data comes from a deterministic source, the “errors” are not random but repeatable for each configuration. These errors can then be described as a further energy function whose values are smoothly varying and correlated. The error within the reference data, $z(\mathbf{x}^i)$, can be modelled by Gaussian functions and sampled points that are similar in the input space have similar errors. Thus the energy for a data point is the sum of two varying functions – the error noise and the deterministic source function. Consequently, the energy prediction can be reformulated as:

$$E(\mathbf{x}^i) = z(\mathbf{x}^i) + \mu. \quad (5)$$

The error is the deviation $z(\mathbf{x}^i)$ of the function value from the global average μ , which is provided by the deterministic source and also includes the error due to noise in the sampled data. This is now expressed by the GPR. The required correlation is constructed from Gaussian, one for each of the known reference structures. The energy prediction then relies on these Gaussian functions, where the contribution by each function depends upon the correlation between the reference configurations and the trial configuration. Given that two similar input vectors \mathbf{x}^i and \mathbf{x}^j have comparable errors, this similarity can be represented by a correlation matrix \mathbf{C} with elements

$$C_{ij} = \text{Corr}(z(\mathbf{x}^i), z(\mathbf{x}^j)) = \exp\left(-\sum_{h=1}^{N_{\text{dim}}} \theta_h |x_h^i - x_h^j|^{p_h}\right), \quad (6)$$

θ_h and p_h are two parameters of the model with $\theta_h \geq 0$ and $1 \leq p_h \leq 2$. Fitting the model then relies on a) the determination of the variance σ^2 of the Gaussians, and b)

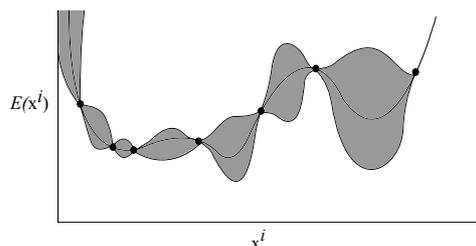


Fig. 1 A one-dimensional interpolation by a Gaussian Process Regression (GPR). The grey region represents the uncertainty of the model for a given input x , which is small near or at known reference points (black dots). The solid curve is the GPR interpolation of the PES. The likelihood is then low in those grey regions with large widths, and so sampling of new training points is directed towards these configurations.

the fitting of the parameter vectors of θ and \mathbf{p} . Then, σ^2 , θ and \mathbf{p} are found by the maximization of the likelihood function L , with these parameters modulating how correlated the Gaussian process values are over the coordinate space. The likelihood function describes the probability that the parameters, σ^2 , θ and \mathbf{p} , determine the PES, and with the vector ϵ containing all N_{ref} reference energies $\epsilon = [E(\mathbf{x}^i), i = 1, 2, \dots, N_{\text{ref}}]^T$, we have:

$$L(\theta, \mathbf{p}, \sigma^2) = \frac{1}{(2\pi)^{N_{\text{ref}}/2} (\sigma^2)^{N_{\text{ref}}/2} |\mathbf{C}|^{1/2}} \times \exp \left[-\frac{(\epsilon - \mathbf{I}\mu)^T \mathbf{C}^{-1} (\epsilon - \mathbf{I}\mu)}{2\sigma^2} \right]. \quad (7)$$

\mathbf{I} is a column vector of 1's, and σ^2 can be expressed with respect to θ and \mathbf{p} . Thus the likelihood function is maximized with respect to θ and \mathbf{p} , and typically it is the log of the likelihood function that is maximized, given $\frac{\partial \log L}{\partial (\sigma^2)} = 0$. In most applications the elements of \mathbf{p} are set to 2, further simplifying the fitting of the likelihood parameters.

New reference structures are added to the data set by determining the likelihood function for unseen examples. Those with the worst likelihood are then used as new reference points. The GPR is iteratively improved by adding more examples to the reference set, with the aim to improve the likelihood function for all reference and unseen structures. Ideally a model should yield high likelihood function values for inputs within the data set, but which has been constructed using the least amount of Gaussians in order to ensure simplicity and optimum generalization properties when applied to new configurations. This is particularly useful if the amount of data is limited, such as in the case of using high-level electronic structure methods to generate the reference data (Fig. 1).

GPRs have already been shown to be applicable to similar types of problems as neural networks, and specifically, in the work of the Popelier group GPRs have been compared to neural networks for modelling environment-dependent atom-centered electrostatic multipole moments [33]. It has been found that in this application GPRs can be even superior to neural networks, with almost a 50% decrease in the electrostatic interaction energy error for the pentamer water cluster [33].

In the following years, the use of GPRs in the Popelier group has been expanded, and been used for the prediction of the electrostatic multipole moments for ethanol [34] and alanine [35], as well as the simulation of hydrated sodium ions [36], and the prediction hydrogen bonded complexes [37].

Bartók et al. have used GPRs to construct PESs of a variety of condensed systems [38]. In this very accurate approach, the total energy is expressed as a sum of environment-dependent atomic energies and provided as a superposition of Gaussians centered at known reference points. A central aspect of the method is the use of four-dimensional spherical harmonics to provide a structural characterization, which is invariant with respect to rotation and translation of the system [38–40]. Their application of GPRs has been either to model the entire PES, like for carbon, silicon, germanium, iron, and GaN [38], or to provide a correction to a DFT PES [40]. In the latter application very accurate energies for water can be obtained by first calculating the energy at the DFT-BLYP level of theory [41,42]. The GPR model is then fitted in order to correct the one and two-body interactions, using more accurate data, e.g. from coupled cluster calculations. Thus a more computationally affordable method can produce high accuracy results without having the need to explicitly compute the energies from an unaffordable high level of theory.

Gaussian functions have also been employed by Rupp et al. [43,44] to fit the atomization energies of organic molecules. For the structural description of a wide range of molecules a Coulomb matrix has been used. While this method does not yet represent a PES suitable for atomistic simulations, as only the global minimum structure of each molecule can be described and as no forces are available, this very promising technique has certainly the capabilities to become a serious alternative approach in the years to come. A first step in this direction has been taken very recently in a first study of conformational energies of the natural product Archazolid A [45].

2.3 Modified Shepard interpolation

In 1994 Ischtwan and Collins developed a technique to construct the PESs of small molecules based on the modified Shepard interpolation (MSI) method [46]. The basic idea is to express the energy of an atomic configuration \mathbf{x} as a sum of weighted second order Taylor expansions centered at a set of reference points known from electronic structure calculations. Typically, a set of $3N_{\text{atom}} - 6$ inverse interatomic distances is used to describe the atomic configuration \mathbf{x} .

Then, for each reference point i , the energy of a configuration \mathbf{x} in its close environment is approximated by a second order Taylor series $E_i(\mathbf{x})$ as:

$$E_i(\mathbf{x}) = E(\mathbf{x}^i) + (\mathbf{x} - \mathbf{x}^i)^T \mathbf{G}_i + \frac{1}{2} (\mathbf{x} - \mathbf{x}^i)^T \mathbf{H}_i (\mathbf{x} - \mathbf{x}^i) + \dots, \quad (8)$$

where \mathbf{G}_i is the gradient and \mathbf{H}_i is the matrix of second derivatives, i.e., the Hessian, at the position \mathbf{x}^i of the reference point.

Typically, many reference points are available, and the energy prediction for an unknown configuration can be improved significantly by employing a weighted sum of the Taylor expansions centered at all reference points,

$$E(\mathbf{x}) = \sum_{i=1}^{N_{\text{ref}}} w_i(\mathbf{x}) \cdot E_i(\mathbf{x}). \quad (9)$$

The normalized weight w_i of each individual Taylor expansion is given by:

$$w_i(\mathbf{x}) = \frac{v_i(\mathbf{x})}{\sum_{k=1}^{N_{\text{ref}}} v_k(\mathbf{x})} \quad (10)$$

and can be determined from the unnormalized weight function

$$v_i(\mathbf{x}) = \frac{1}{|\mathbf{x} - \mathbf{x}^i|^p}, \quad (11)$$

which approaches zero for very distant reference points and infinity for $\mathbf{x} = \mathbf{x}^i$ corresponding to a normalized weight $w_i(\mathbf{x}) = 1$. The power parameter p needs to be equal or larger than the order of the Taylor expansion. The obtained PES has a number of favorable features. It depends only on interatomic distances, and therefore it is translationally and rotationally invariant. Further, the PES is also invariant with respect to a permutation of chemically equivalent atoms.

The accuracy of the method crucially depends on the choice of the reference points, as the approximation of a second order Taylor expansion of the energy is only sufficiently accurate, if there is a dense set of reference structures. It has been suggested by Ischtwan and Collins [47] to start with a reasonable first set of structures, which can be chosen along a reaction path, and to improve the quality of the PES by adding more structures in chemically relevant parts of the configuration space. These points are identified by running classical molecular dynamics trajectories employing preliminary potentials, which are then iteratively refined by adding more and more points from electronic structure calculations carried out at the configurations visited in these trajectories.

The modified Shepard interpolation scheme has been applied successfully to study chemical reaction dynamics for a number of systems, like $\text{NH} + \text{H}_2 \rightarrow \text{NH}_2 + \text{H}$ [47], $\text{OH} + \text{H}_2 \rightarrow \text{H}_2\text{O} + \text{H}$ [48,49], and many others [50–53]. It has also been adapted to describe molecule-surface interactions using the dissociation of H_2 at the Pt(111) surface as an example by Crespos et al. [54,55]. To keep the dimensionality and thus the complexity of the system at a reasonable level, the positions of the surface atoms have been frozen resulting in a six-dimensional PES. As the hydrogen atoms can move along the surface, an extension of equation (12) has been required to include the symmetry of the surface. The modified energy expression is:

$$E(\mathbf{x}) = \sum_{i=1}^{N_{\text{ref}}} \sum_{j=1}^{N_{\text{sym}}} w_{ij}(\mathbf{x}) \cdot E_i(\mathbf{x}). \quad (12)$$

The purpose of the second summation over all symmetry elements N_{sym} ensures that the full symmetry of the surface is taken into account exactly without the requirement to add structures, which are equivalent by symmetry, into the reference set.

The modified Shepard interpolation is very appealing because of its simplicity and the accuracy of the obtained PESs, but it has also a number of drawbacks. First, to date its applicability is limited to very low-dimensional systems and, once the PES has been constructed, the system size cannot be changed, because if an atom would be added, the distance to the reference points in the Taylor expansions would not be defined. Moreover, for the prediction of the energy of a new configuration the reference set needs to be known and a summation of all Taylor expansions is required, which makes the method more demanding for large reference sets. Further, the construction of the potential requires energies, gradients and Hessians from electronic structure calculations, which makes the determination of the reference data rather costly.

Given the similarities between Shepard interpolation and polynomial fitting in terms of applicability it is worth comparing these methods in terms of how much data is required to achieve a fit for the CH_5^+ system. Huang et al. made use of 20 728 training energies to fit a polynomial surface [30]. However, Wu et al. have been able to achieve a similar fit using just 50 training examples employing MSI [50,51]. However, these 50 training examples are not just energies, but include forces and Hessians. In order to determine these Hessians, numerous energy calculations must be performed, resulting in 19 440 energy calculations – a comparable number to that of Huang et al.

2.4 Interpolating moving least squares

The standard implementation of interpolating moving least squares (IMLS) [56,57], which is a method derived from the modified Shepard interpolation, determines a configuration's energy using a number of linearly independent basis functions, which can be represented by a matrix. The energy is then given as the sum of M basis functions b_i of the atomic coordinates \mathbf{x} , where each basis function is weighted by a coefficient a_i ,

$$E(\mathbf{x}) = \sum_{i=1}^M a_i(\mathbf{x}) b_i(\mathbf{x}). \quad (13)$$

The basis functions have either the form of a Taylor series [57] or a many-body expansion similar to those used in high-dimensional model representation neural networks [58]. The coefficients are determined by minimization of the objective function

$$D[E(\mathbf{x})] = \sum_{i=1}^{N_{\text{ref}}} w_i(\mathbf{x}) \left[\sum_{j=1}^M a_j(\mathbf{x}) b_j(\mathbf{x}) - E(\mathbf{x}^i) \right]^2, \quad (14)$$

which is the weighted sum of squared potential energy errors, with $w_i(\mathbf{x})$ being the weight defined by the proximity of the trial structure to a reference structure i .

Hence, the largest weight would be found if the trial configuration coincides with a configuration in the reference set. The method is easily expanded to include further fitting data, such as forces and the Hessians of the training points [58,59], which expands the design matrix – the matrix containing the simultaneous equations that must be solved in order to find the best combination of basis functions. The coefficients a_j are determined by solving:

$$\mathbf{B}^T \mathbf{W} \mathbf{B} \mathbf{a} = \mathbf{B}^T \mathbf{W} \mathbf{V}. \quad (15)$$

Here \mathbf{V} is the vector of reference energies, and \mathbf{a} is the transpose of the vector containing the coefficients. \mathbf{W} is a $N_{\text{ref}} \times N_{\text{ref}}$ diagonal matrix containing the weights, where $W_{ij} = w_i(\mathbf{x})\delta_{ij}$. \mathbf{B} is the $N_{\text{ref}} \times M$ design matrix.

Initial applications of the IMLS method focused on fitting available PESs for HN_2 [60], HOOH [58,61], and HCN [58], before moving on to PESs computed using accurate ab initio methods, such as for HOOH , CH_4 , and HCN [59], CH_2 and HCN [62], as well as the $\text{O} + \text{HCl}$ system [63]. More recently Dawes et al. have applied IMLS fitted PES to van der Waals systems, including the CO [64], NNO [65] and OCS [66] dimers, described as four-dimensional PESs where data points have been obtained from CCSD(T) calculations. This method of data generation is suitable for IMLS as accurate fits can be constructed from relatively small amounts of data. This level of accuracy is often required, e.g. for the prediction of rovibrational spectra.

The basis functions used by Dawes et al. take the form of numerous many-body terms, that together form a high-dimensional model representation [67]. High order terms are not included as basis functions as they would be poorly determined in scenarios where the PES has been sparsely sampled.

In order to perform the fitting of the HOOH PES, Dawes et al. used and iteratively sampled training set which eventually included almost 2000 reference points [58]. However, compared to Shepard interpolation models and neural networks, IMLS models are costly to evaluate. Dawes et al. showed that IMLS can be used as a procedure to create an accurate PES from a sparse set of reference data points. This IMLS PES is then employed to generate training points that are in turn used to train a Shepard interpolation model and a neural network.

Using two IMLS models, where one uses basis functions that are an order of degree higher than those used in the other model, can guide the sampling of the PES. Regions with a large error between the two models, which will occur between the current training points, will be the ideal locations to sample and in turn improve the model. In this way the manner in which the PES is sampled is similar to GRPs, where the sampling is sparse while non-stochastic in nature [59]. The final model used by Dawes et al. in this example is a modification of IMLS, a local-IMLS [68], which is similar to the Shepard interpolation scheme. Local-IMLS computes energies for a configuration by using the reference structures that are similar rather than using all reference configurations. However, local-IMLS is more general since the basis functions can

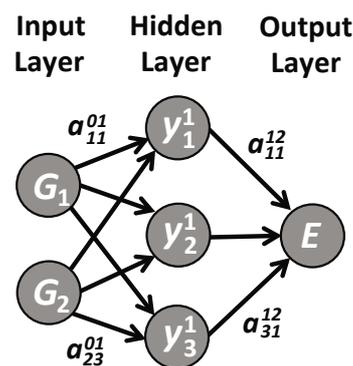


Fig. 2. A small two-dimensional feed-forward Neural Network (NN) containing a single hidden layer. The input nodes G_1 and G_2 provide the structural information to the NN, which then yields the energy E . Apart from the architecture of the NN, i.e., the number of layer and the number of nodes per layer, the NN output is determined by the numerical values of the weight parameters, which are shown as arrows indicating the flow of information through the NN. The functional form of this NN is given in equation (17).

take any form, rather than just the Taylor series expansions used in Shepard interpolation. For the HOOH PES a similar accuracy compared to previous work was achieved using 30% less reference points.

2.5 Neural network potentials

Artificial neural networks (NNs) [69,70] have first been introduced already in 1943 to develop mathematical models of the signal processing in the brain [71], and after continuous methodical extensions in the following decades they have now become an established research topic in mathematics and computer science with many applications in various fields. Also in chemistry and physics they have found wide use, e.g., for the analysis of spectra and pattern recognition in experimental data [72]. The use of NNs to construct PESs employing reference data from electronic structure calculations has been suggested by Blank et al. [73] almost twenty years ago, and in the meantime neural network potentials (NNPs) have been reported for a number of molecules [74–76] and molecules interacting with surfaces [77–79]. A comprehensive overview about the systems that have been addressed by NNPs can be found in several recent reviews [80–82].

The basic component of almost any NNP is a feed-forward NN, and a small example corresponding to a two-dimensional PES is shown in Figure 2. It consists of artificial neurons, or nodes, which are arranged in layers. The neurons in the input layer represent the coordinates G_i defining the atomic configuration, the neuron in the output layer yields the energy of this configuration. In between the input and the output layer there are one or more so-called hidden layers, each of which contains a number of additional neurons. The purpose of

the hidden layers, which do not have a physical meaning, is to define the functional form of the NN assigning the energy to the structure. The more hidden layers and the more nodes per hidden layer, the higher is the flexibility of the NN. Haykin has stated that in the case of a NN architecture consisting of two hidden layers, that the first hidden layers capture the local features of the function being modelled, while the second hidden layer capture the global features of the function [83]. All nodes in all layers are connected to the nodes in the adjacent layers by so-called weight parameters, which are the fitting parameters of the NN. In Figure 2 they are shown as arrows indicating the flow of information through the NN. The value y_i^j of a node i in layer j is then calculated as a linear combination of the values of the nodes in the previous layer using the weight parameters as coefficients. The linear combination is then shifted by a bias weight b_i^j acting as an adjustable offset. Finally, a non-linear function, the activation function of the NN, is applied, which provides the capability of the NN to fit any real-valued function to arbitrary accuracy. This fundamental property of NNs has been proven by several groups independently [84,85] and is the theoretical basis for the applicability of NNs to construct atomistic potentials. Accordingly, y_i^j is then obtained as:

$$y_i^j = f \left(b_i^j + \sum_{k=1}^{N_{j-1}} a_{k,i}^{j-1,j} \right), \quad (16)$$

where $a_{k,i}^{j-1,j}$ is the weight parameter connecting node k in layer $j-1$ containing N_{j-1} neurons to node i in layer j . In the special case of the input layer with superscript 0, the input coordinates are represented by a vector $\mathbf{G} = \{G_i\} = \{y_i^0\}$. The complete total energy expression of the example NN shown in Figure 2 is then given by:

$$E = f \left(b_1^2 + \sum_{j=1}^3 a_{j1}^{12} \cdot f \left(b_j^1 + \sum_{i=1}^2 a_{ij}^{01} \cdot G_i \right) \right). \quad (17)$$

The weight parameters of the NN are typically determined by minimizing the error of a training set of electronic structure data employing gradient-based optimization algorithms, and in particular the backpropagation algorithm [86] and the global extended Kalman filter [87] have been frequently used in the context of NNPs. The target quantity is usually the energy, but also forces have been used in some cases [88,89].

NNPs constructing the potential-energy using just a single feed-forward NN have been developed for a number of systems with great success, but for a long time NNPs have been restricted to low-dimensional PESs involving only a few degrees of freedom. This limitation has a number of reasons. First of all, each additional atom in the system introduces three more degrees of freedom, which need to be provided to the NN in the input layer. Increasing the number of nodes in the NN reduces the efficiency of the NN evaluation, but it also complicates the determination of the growing number of weight parameters. Further,

conventional NNPs can only be applied to systems with a fixed chemical composition, because the number of input nodes of the NN cannot be changed once the weight parameters have been determined. If an atom would be added, the weight parameters for the additional input information would not be available, while upon removing an atom the values of the corresponding input nodes would be ill-defined.

The most severe challenge of constructing NNPs is the choice of a suitable set of input coordinates. Cartesian coordinates cannot be used, because their numerical values have no direct physical meaning and only relative atomic positions are important for the potential-energy. If, for instance, a molecule would be translated or rotated in space, its Cartesian coordinates would change while its internal structure and thus its energy are still the same. An NNP based on Cartesian coordinates, however, would predict a different energy since its input vector has changed. Incorporating this rotational and translational invariance by a coordinate transformation onto a suitable set of functions exhibiting these properties is a significant challenge. A very simple solution appropriate for small molecules would be to use internal coordinates like interatomic distances, angles, and dihedral angles, and indeed this has been done successfully for a number of molecules, but this approach is unfeasible for large systems due to the rapidly increasing number of coordinates. Further, a full set of internal coordinates contains a lot of redundant information, and also using a full-distance matrix is possible only for small systems.

Another problem that is still present even if internal coordinates are used concerns the invariance of the total energy with respect to the interchange of atoms of the same element. If, for instance, the order of both hydrogen atoms in a free water molecule is switched then this change is also present in the NN input vector. Since all NN input nodes are connected to the NN by numerically different weight parameters, this exchange will modify the total energy of the system, although the atomic configuration is still the same. Including this permutation symmetry of like atoms into NNPs has been a challenge since the advent of NNPs. A solution for molecular systems based on a symmetrization of internal coordinates has been suggested in a seminal paper by Gassner et al. [76] for molecular systems, and a similar scheme has also been developed for the dissociation of molecules at surfaces [78]. Unfortunately, both methods are only applicable to very low-dimensional systems. It should be noted, however, that also low-dimensional NNPs have been applied to study large systems by representing only specific interactions by NNs. Examples are the work of Cho et al. [90], who used NNs to include polarization in the TIP4P water model [91], and the description of three-body interactions in the solvation of Al^{3+} ions by Gassner et al. [76].

It has soon been recognized that further methodical extensions are required to construct high-dimensional NNPs capturing all relevant many-body effects in large systems. A first step in this direction has been made by Manzhos and Carrington [92,93] by decomposing the PESs into

interactions of increasing order in the spirit of a many-body expansion

$$E = \sum_{i=1}^{N_{\text{atom}}} E_i + \sum_i \sum_{j>i} E_{ij} + \sum_i \sum_{j>i} \sum_{k>j} E_{ijk} + \dots, \quad (18)$$

with the E_i , E_{ij} , E_{ijk} being one-, two-, three-body terms and so forth.

Specifically, they have used the high-dimensional model representation of Li et al. [67] to reduce the complexity of the multidimensional PES by employing a sum of mode terms, each of which is represented by an individual NN and depends only on a small subset of the coordinates. Since the number of terms, and consequently also the number of NNs that need to be evaluated, grows rapidly with system size and with the maximum order that is considered, to date the method has only been applied to comparably small molecular systems. In the following years the efficiency could be improved by reducing the effective dimensionality employing optimized redundant coordinates [94,95]. This approach is still the most systematic way to construct NNPs and very accurate PESs can be obtained. A similar method based upon many-body expansions has also been suggested by Malshe et al. [96].

First attempts to develop high-dimensional NNPs have been made by Smith and coworkers already in 1999 by introducing NNs of variable size and by decomposing condensed systems into a number of atomic chains [97,98]. In this early work the NN parameters have not been determined using electronic structure reference data, and only in 2007 the method has been further developed to a genuine NNP approach [99,100] using silicon as example system. Surprisingly, no further applications or extensions of this method have been reported to date.

A high-dimensional NNP method, which has been applied to a manifold of systems, has been developed by Behler and Parrinello in 2007 [101]. In this approach, the energy of the system is constructed as a sum of environment-dependent atomic energy contributions E_i ,

$$E = \sum_i E_i. \quad (19)$$

Each atomic energy contribution is obtained from an individual atomic NN. The input for these NNs is formed by a vector of symmetry functions [102], which provide a structural fingerprint of the chemical environments up to a cutoff radius of about 6–10 Å. The functional form of the symmetry functions ensures that the potential is translationally and rotationally invariant, and due to the summation in equation (19) the potential is also invariant with respect to permutations of like atoms. Consequently, this approach overcomes all conceptual limitations of conventional low-dimensional NNPs. For multicomponent systems long-range electrostatic interactions can be included by an explicit electrostatic term employing environment-dependent charges, which are constructed using a second set of atomic NNs [103,104]. This high-dimensional NNP, which is applicable to very large systems containing thousands of atoms, has been applied to a number of systems

like silicon [105], carbon [106], sodium [107], copper [108], GeTe [109], ZnO [104], the ternary CuZnO system [110], and water clusters [111,112].

NNs have also been employed by Popelier and coworkers to improve the description of electrostatic interactions in classical force fields [33,113–115]. For this purpose they have developed a method to also express higher order electrostatic multipole moments of atoms as a function of the chemical environment. The reference multipole moments used to train the NNs are obtained from DFT calculations and a partitioning of the self-consistent electron density employing the concept of quantum chemical topology (QCT) [116–118]. The neural network acts by taking the input, the relative position of atoms about the atom for which the predictions are made, and then expressing the multipole moments up to high orders within the atomic local frame. The advantage of this method is that all electrostatic effects, such as Coulombic interactions, charge transfer and polarization, which traditionally are represented in force fields using different interaction terms, are instead treated equally since they all are a result of the predicted multipoles. Furthermore this also means that all electrostatic interactions are more realistic because of the non-spherical nature of the atomic electron densities that are described by the multipoles. Polarization is then a natural result of the prediction of these multipoles, as they change as atoms move around, and atomic electron densities deform. It has been demonstrated that the accuracy of electrostatic interactions in classical force fields can be significantly improved, but if high-order multipoles are used, the number of NNs to be evaluated can be substantial.

Recently, NNPs have also been constructed using input coordinates, which have been used before in other types of mathematical potentials. An illustrative example is the use of permutationally invariant polynomials by Li et al. [119], which have been used very successfully by Bowman and coworkers for a number of molecular systems (cf. Sect. 2.1). On the other hand, in the work of Fournier and Slava, symmetry functions similar to those introduced by Behler [102], are used for an atomistic potential, but here a many-body expansion is employed rather than a neural network [120]. Using the symmetry functions as the dimensions of the descriptor space, a clustering procedure can be used to define the atom types. Within the PES, the energy contribution for an atom is then determined from the linear combination of each atom type. In a similar manner, a symmetry function based neural network can also be used for structure analysis in computer simulations [121].

NNs and GPRs show similar limitations, namely their inability to extrapolate accurate energies for atomic configurations being very different from the structures included in the training set. Both methods are thus reliant on the sampling of the training points to guide the fitting of the function [122]. GPRs model smoothly varying functions in order to perform regression and interpolation, and typically the kernels used in most GPR models perform poorly for extrapolation, though there are attempts to address this issue [123].

2.6 Support vector machines

Support vector machines (SVMs) [124] are a machine learning method that has found much use in the fields of bioinformatics and cheminformatics [125,126] mainly for classification problems. However, SVMs can also be used for regression and so are suitable for PES fitting, though examples of such applications of SVMs are still rare.

In general, SVMs aim to define a linear separation of data points in feature space by finding a vector that yields the largest linear separation width between two groups of points with different properties. The width of this separation is given by the reference points that lie at the margin either side of the separating vector and these reference points are called support vectors. A balance needs to be found between finding the maximum separation, and errors, which arise from those cases where data points lie on the wrong side of the separation vector and so are incorrectly classified.

Unfortunately, a linear separation is often impossible in the original input coordinate feature space. This problem can be solved by transforming the input coordinates and recasting them in a higher-dimensional space, where a linear separation becomes possible. Depending on the way this non-linear transformation is performed the linear separation vector found in this high-dimensional space can then be transformed back into a non-linear separator in the original coordinate space.

There are many ways to perform the transformation from the original input space into the higher dimensional feature space. One possibility is to define the dimensions of the higher-dimensional space as combinations of the original coordinates. This feature space could then in theory contain an infinite number of dimensions. The classification $a(\mathbf{x})$, of a trial point \mathbf{x}^i , is then given by the general form of a SVM

$$a(\mathbf{x}) = \sum_{i=1}^{N_{\text{ref}}} \lambda_i y_i \mathbf{K}(\mathbf{x}^i, \mathbf{x}) - w_0. \quad (20)$$

The sum is over the number of support vectors, which is equal to the number of reference examples. \mathbf{K} is the kernel that transforms the input, \mathbf{x} , from the original representation to the new higher order representation and compares the similarity between the trial point and each of the reference points.

Equation (20) can also be viewed as a representation of a neural network by a SVM, where w_0 would be the bias of the hidden layer. This means that y_i is the classification of reference example \mathbf{x}^i , and λ_i is a coefficient. The elements of \mathbf{K} are found by:

$$\mathbf{K}(\mathbf{x}^i, \mathbf{x}) = \tanh(k_0 + k_1 \langle \mathbf{x}^i, \mathbf{x} \rangle). \quad (21)$$

k_0 is the bias weight of the first layer of the neural network. Thus $k_1 \langle \mathbf{x}^i, \mathbf{x} \rangle$ is the dot product of the vectors, and so is a measure of similarity. $\mathbf{K}(\mathbf{x}^i, \mathbf{x})$ in effect represents the action of the first layer of hidden nodes of a neural network, as it applies a hyperbolic tangent function to $k_1 \langle \mathbf{x}^i, \mathbf{x} \rangle$.

A SVM is more general than a neural network, since different kernels allow for other non-linear transformations to be used. Without the non-linear transformation, each element of the kernel can be viewed as a measure of similarity between the reference points and the trial points. If we perform the non-linear transformation to the kernel we have two approaches. The first is to explicitly compute the new variables of the vector with respect to the new high-dimensional space. While this is trivial if the number of dimensions is small, the computation will grow in complexity, quadratically, as more dimensions are added, to the point where the computation will not fit within the computer memory. A solution to this problem is the so-called “kernel trick”, which accounts for the non-linear transformation into a high-dimensional space, but where the original dot product matrix is still used. The transformation is then applied after finding the dot products. Thus the non-linear transformation is implicit, and simplifies the whole process, since this kernel is integral to then finding the linear separation vector.

The main task to be solved when using SVMs is finding the linear separator so that the distance of separation is maximized, the errors in classification are reduced, and identifying the appropriate non-linear transformation, or kernel function, that yields the reference data in a new high-dimensional feature space, where the separation can be performed.

Since there is an infinite number of possible non-linear transformations, and the use of multiple kernels can allow for multiple classifications to be separated, it can happen that the flexibility offered by SVMs results in overfitting and a loss of generalization capabilities. For this reason, like in many other fitting methods, cross validation is applied during fitting.

The linear separation in high-dimensional space is then defined by:

$$\mathbf{w} \cdot \mathbf{X} - b = 0, \quad (22)$$

where \mathbf{X} is a set of reference points that define the linear separator, and \mathbf{w} is the normal vector to the linear separator. Furthermore the margins of the separation can be defined by:

$$\mathbf{w} \cdot \mathbf{X} - b = -1 \quad (23)$$

and

$$\mathbf{w} \cdot \mathbf{X} - b = 1. \quad (24)$$

as shown in Figure 3.

The width of the separation is given by $\frac{2}{\|\mathbf{w}\|}$. The objective of a SVM is then to maximize this separation. The constraint of the linear separation can be rewritten as:

$$y_i (\mathbf{w} \cdot \mathbf{x}^i - b) \geq 1, \quad (25)$$

where \mathbf{x}^i is reference training point that must be correctly classified, and y_i is the classification function, i.e., the output is then either 1 or -1 . This can be modified to allow for soft margins, and so allow for a degree of misclassification

$$y_i (\mathbf{w} \cdot \mathbf{x}^i - b) \geq 1 - \xi^i. \quad (26)$$

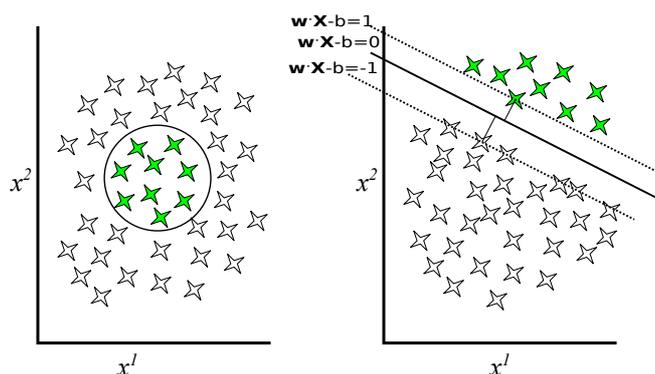


Fig. 3. In the initial two-dimensional space that data points (stars) fall into two groups – those within the circle, and those outside of it. Within this space there is no way to perform a linear separation. A kernel allows to recast the data points into a new space where a linear separation is possible. Those within the circle have classification ‘1’, and those outside have classification ‘-1’. The linear separation is defined by a vector, with a gradient of \mathbf{w} , and an intercept of b . The linear separation is found as vector that gives the widest separation between the two classes. This width is determined by two data points, the support vectors.

ξ^i is called the slack variable, a measure of the degree of misclassification. This is a penalty that must be minimized, as part of the objective function J ,

$$\min J(\mathbf{w}, \xi) = \frac{1}{2} \mathbf{w}^T \mathbf{w} + c \sum_{i=1}^{N_{\text{ref}}} \xi^i, \quad (27)$$

where c is a parameter modulating the influence of the slack function.

For regression purposes like the construction of PESs, the objective is the reverse problem, where the reference data points should lie as close as possible to a linear separation. This is the basis of Least-Squares Support Vector Machines (LS-SVMs) and their applications to PESs in the work of Balabin and Lomakina [127].

LS-SVMs require the minimization of

$$\min J(\mathbf{w}, b, e) = \frac{\nu}{2} \mathbf{w}^T \mathbf{w} + \frac{\zeta}{2} \sum_{i=1}^{N_{\text{ref}}} e_i^2. \quad (28)$$

The slack values are now replaced by the sum of squared errors e_i of the reference points, and ν , and ζ , are parameters that determine the smoothness of the fit. Both SVMs and GPRs are similar in that they can both be adjusted by the choice of the kernel function used. Gaussians are frequently used as they are computationally simple to implement, but other choices of functions can adjust the sensitivity of the fitting method [128].

Specifically, Balabin and Lomakina used LS-SVMs to predict energies of calculations with converged basis sets employing electronic structure data obtained using smaller basis sets for 208 different molecules in their minimum geometry containing carbon, hydrogen, oxygen, nitrogen, and fluorine. Therefore, while not describing the PES for

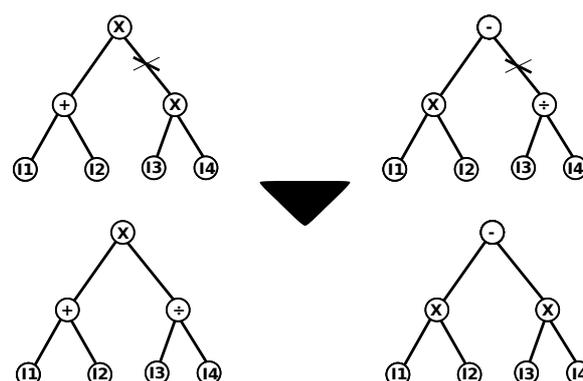


Fig. 4. The two parent Genetic Programming (GP) trees at the top, have been randomly partnered, and at the same position on each tree the connection is marked by an ‘x’ where the branches will be exchanged. At the end of the branches are the input nodes, I1-I4, which provide information about the system. At each node on the way up the tree an arithmetic operation is performed that combines the values passed up from the lower level. This process continues until the last node has been reached, where the output is determined.

arbitrary configurations this approach is still an interesting example of machine learning methods applied to energy predictions, and an extension to the representation of PESs would be possible. In comparison to neural networks, the LS-SVM model required less reference points, and provided slightly better results. Very recently, SVMs have been used for the first time to construct a continuous PES by Vitek et al. [129], who studied water clusters containing up to six molecules.

2.7 Genetic programming – function searching

Genetic programming (GP) uses populations of solutions, analyzes them for their success, and employs, similar to genetic algorithms, survival of the fittest concepts as well as mating of solutions and random mutations to create new populations containing better solutions [130]. It is based upon the idea of a tree structure (cf. Fig. 4). At the very base of the tree are the input nodes which feed in information about the configuration of the atomic system. As one moves up the tree, two inputs from a previous level are combined returning an output, which is fed upwards the tree to the next node and so forth until in the uppermost node a final arithmetic procedure is performed and returns the output, the energy prediction. In this manner complex functions can be represented as a combination of simpler arithmetic functions.

Genetic Programming in the above manner then relies on the randomization of the functions in all members of the population. New daughter solutions are generated by randomly performing interchanges between two members of the best members of the parent population. The interchange occurs by randomly selecting where in the tree structure for each parent the interchange will occur. This means that the tree structure can be very different for the daughter solutions, with the tree structure having been

extended or pruned. Randomly, the daughter solutions may also undergo mutations, i.e., a change in a function used in the tree structure, or a change in a parameter. Many other types of crossover of tree structures can also be used. Depending on the choice of functions, GPs can have the advantage that they can provide human readable and sometimes even physically meaningful functions, in contrast to most methods discussed above.

Genetic programming has been applied to simple PESs, such as that of a water molecule [131], though more complex formulations of GPs have been used by Bellucci and Coker [132] to represent an empirical valence bond potential formulation for determining the energy change in reactions. Here, three PESs have to be found, one of the products, one for the reactants, and one that relates the two surfaces and describes the reaction. Bellucci and Coker build on the concept of directed GPs, where the functional form of the PES is partially defined, and the GP search modifies and improves this first potential. For example the GP searches for the best function that performs a non-linear transformation on the inputs before they are passed to the main predefined function. The aim is to find the best Morse and Gaussian functions that together describe the reactions. They go beyond traditional GP in the use of multiple GPs, where a higher level of GP modifies the probabilities of the search functions that direct the lower level GP fitting the PES functions.

Brown et al. proposed a hierarchical approach to GP where the GP population is ranked based upon their fitness [133]. This allows for the training procedure to retain individuals so that, while convergence is achieved, it is not at the cost of the diversity of the solutions. Maintaining a diverse set of solutions for the PES fitting allows for the generation of better approximations to the global minimum of the target function, rather than just converging to a local minimum. The ranking of solutions is similar to the concept of multi-objective genetic algorithms and the use of Pareto ranking [134].

3 Discussion

All the potentials discussed in this review have in common that they are employing very general functional forms, which are not based on physical considerations. Therefore, they are equally applicable to all types of bonding and in this sense they are “non-physical”, i.e., they are completely unbiased. In order to describe PESs correctly, the information about the topology of the PESs needs to be provided in form of usually very large sets of reference electronic structure calculations, and the presented potentials differ in the manner this information is stored. For some of the potentials it is required that this data base is still available when the energy of a new configuration is to be predicted. This is the case for the MSI, IMLS, and GPRs making these methods more demanding with growing reference data set size. On the other hand, these methods reproduce the energies of the reference structures in the final potentials error-free. Methods like NNPs and

permutation invariant polynomials transform the information contained in the reference data set into a substantial number of parameters, and the determination of a suitable set of parameter values can be a demanding task as far as computing time is concerned. Still, in contrast to many conventional physical potentials, there is usually no manual trial-and-error component in the fitting process. Often, like in case of NNPs, gradient-based optimization algorithms are used, but they bear the risk of getting trapped in local minima of the high-dimensional parameter space, and there is usually no hope to find the global minimum. Fortunately, in most cases, sufficiently accurate local minima in parameter space can be found, which yield reliable potentials.

An interesting alternative approach to find the parameters, which has been used a lot in the context of potential development, but not for the potentials above, is to employ genetic algorithms (GAs) [135]. Like neural networks they represent a class of algorithms inspired by biology and belong to the group of machine learning techniques. GAs work by representing the parameters determining a target value as a bit string. This bit string is randomized and used to generate an initial population of bit strings that represent a range of values for the parameters. Thus, the bit string represents not one, but a series of parameter values, where each member of the population has different values for these parameters, but each parameter can only vary within given thresholds. Each member of the initial population is assessed for its performance. The best performing members of the population are then used to generate the next “daughter population”. This is done by randomly selecting where along the bit string two population members undergo an interchange. The result is that from two “parent” bit strings two daughter bit strings are obtained. The central idea is that one of the daughter bit strings will retain the best parameters of both parents. In order to allow for exploration of the parameter space, the daughters may also randomly undergo a “mutation”. A mutation is where a random bit in the daughter bit string is switched from 0 to 1, or vice versa. The final stage is to combine the daughter and parent populations, and from the combined population, remove the poorly performing bit strings. The final population then should be a combination of the previous population and the newer daughter population. From this new population, the process can be started again, and over a number of generations the population should converge on a bit string that gives the best parameter set. GAs have found wide use for many problems in chemistry and physics, such as the searching the conformation space [136,137] and protein folding [138], as well as the assignment of spectral information [139,140].

GAs have also been used in many cases for the determination of parameters in atomistic potentials. The aim is to find a set of parameters that minimizes a fitness function, where the fitness is given as a sum of weighted squared errors. For instance, the work of Marques et al. aims to simultaneously fit the potential, in their case the extended-Rydberg potential, for NaLi and Ar₂ to both ab initio energies and to spectral data [141]. Similarly, in the work

of Da Cunha et al. [142] 77 parameters have been fitted for the Na + HF PES, and Roncaratti et al. [143] used GAs to find the PESs of H_2^+ and Li_2 . Xu and Liu use a GA to fit the EAM model for bulk nickel where six parameters have been varied [144]. Rather than fitting parameters, a GA can also be used to identify energetically relevant terms from a large number of possible interactions, which has been suggested for the cluster expansion method in the approach taken by Hart et al. [145,146]. The danger in all of these cases is that the GA will yield a single solution, and that this solution represents the optimization of the objective function to a local minimum. Therefore there is the need to rerun the fitting procedure from different starting points, i.e., different positions on the PES fitting error surface, in order to not get trapped in local minima.

Parameter fitting for force fields has also been investigated by Pahari and Chaturvedi [147], Larsson et al. [148], and Handley and Deeth [149]. In the work of Larsson et al. GAs are used to automate the exploration of the multidimensional parameter space of the ReaxFF force field, where 67 parameters are to be optimized for the prediction of the PES of SiOH. Pahari and Chaturvedi perform a GA fitting of 51 of the ReaxFF parameters, which were determined to be important to the description of CH_3NO . In the work of Handley and Deeth the aim has been to parameterize the Ligand Field Molecular Mechanics (LFMM) force field for the simulation of iron amine complexes [149]. LFMM augments standard force fields with terms that allow for determination of the influence of d -orbital electrons on the PES. Here a multi-objective GA approach has been implemented, where Pareto front ranking is performed upon the population [134].

GAs have not yet been employed to find parameters for the potentials reviewed above, and certainly further methodical extensions are required to reduce the costs of using GAs for determining thousands of parameters, e.g., in case of NNPs. The reason for these high costs is that the determination of the fitness function requires performing gradient-based optimizations of the parameters for each member of the population in each generation, since mutations or crossovers for the generation of new population members typically results in non-competitive parameter sets that need to be optimized to find the closest local minimum.

4 Conclusions

A variety of methods has now become available to construct general-purpose potentials using bias-free and flexible functional forms enabling the description of all types of interactions on an equal footing. These developments have been driven by the need for highly accurate potentials providing a quality close to first-principles methods, which is very difficult to achieve by conventional atomistic potentials based on physical approximations. The selection of the specific functional forms of these “next-generation potentials” has been guided by the physical problems to be studied, and there are methods aiming primarily at low-dimensional systems like small molecules and molecule-

surface interactions, e.g. polynomial fitting, the modified Shepard interpolation, and IMLS. Other methods have focussed on condensed systems and are able to capture high-order many-body interactions as needed for the description of metals, like NNPs and GPRs. The use of further methods, like support vector machines and genetic programming, for the construction of potentials is still less evolved, but this will certainly change in the near future.

All of these potentials have now overcome the conceptual problems related to incorporating the rotational and translational invariance as well as the permutation symmetry of the energy with respect to the exchange of like atoms. Most of the underlying difficulties rather concerned the geometrical description of the atomic configurations than the fitting process itself. A lot of interesting solutions have been developed for this purpose, which should in principle be transferable from one method to another. The current combinations of these quite modular components, i.e., describing the atomic configuration and assigning the energy, have been mainly motivated by the intended applications. Therefore, it is only a question of time when ideas will spread and concepts from one approach will be used in other contexts to establish even more powerful techniques.

At the present stage, it has been demonstrated for many examples covering a wide range of systems, from small molecules to solids containing thousands of atoms, that a very high accuracy of typically only a few meV per atom with respect to a given reference method can be reached. However, due to the non-physical functional form, large reference data sets are required to construct these potentials making their development computationally more demanding than for most conventional potentials. Still, this effort pays back quickly in large-scale applications, which are simply impossible otherwise. However, the number of applications, which clearly go beyond the proof-of-principle level by addressing and solving physical problems that cannot be studied by other means, is still moderate. Most of these applications can be related to a few very active research groups, which are involved in the development of these methods and have the required experience. This is because presently the complexity of the methods and of the associated fitting machinery is a major barrier towards their wider use, but again, it is only a question of time when more program packages will become generally available.

In summary, tremendous progress has been made in the development of atomistic potentials. A very high accuracy can now be achieved, and it can be anticipated that these methods will be further extended and become established tools in materials science. Some challenges remain, like the construction of potentials for systems containing a very large number of chemical elements, which are difficult to describe geometrically and rather costly to sample when constructing the reference sets due to the size of their configuration space. On the other hand, to date only a tiny fraction of the knowledge about machine learning methods, which is available in the computer science community, has been transferred to the construction

of atomistic potentials. Consequently, many exciting new developments are to be expected in the years to come.

Financial support by the DFG (Emmy Noether group Be3264/3-1) is gratefully acknowledged.

References

- R.G. Parr, W. Yang, *Density Functional Theory of Atoms and Molecules* (Oxford University Press, 1989)
- R. Car, M. Parrinello, *Phys. Rev. Lett.* **55**, 2471 (1985)
- D. Marx, J. Hutter, *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods* (Cambridge University Press, 2009)
- A.P. Sutton, M.W. Finnis, D.G. Pettifor, Y. Ohta, *J. Phys. C* **21**, 35 (1988)
- C.M. Goringe, D.R. Bowler, E. Hernandez, *Rep. Prog. Phys.* **60**, 1447 (1997)
- M. Elstner, *Theor. Chem. Acc.* **116**, 316 (2006)
- L. Colombo, *Comput. Mater. Sci.* **12**, 278 (1998)
- T. Hammerschmidt, R. Drautz, D.G. Pettifor, *Int. J. Mater. Res.* **100**, 1479 (2009)
- N. Allinger, in *Advances in Physical Organic Chemistry* (Academic Press, 1976), Vol. 13, pp. 1–82
- B.R. Brooks, R.E. Bruccoleri, B.D. Olafson, D.J. States, S. Swaminathan, M. Karplus, *J. Comput. Chem.* **4**, 187 (1983)
- A.C.T. van Duin, S. Dasgupta, F. Lorant, W.A. Goddard III, *J. Phys. Chem. A* **105**, 9396 (2001)
- A. Warshel, R.M. Weiss, *J. Am. Chem. Soc.* **102**, 6218 (1980)
- J. Tersoff, *Phys. Rev. Lett.* **56**, 632 (1986)
- J. Tersoff, *Phys. Rev. B* **37**, 6991 (1988)
- D.W. Brenner, *Phys. Rev. B* **42**, 9458 (1990)
- E. Pijper, G.-J. Kroes, R.A. Olsen, E.J. Baerends, *J. Chem. Phys.* **117**, 5885 (2002)
- M.S. Daw, S.M. Foiles, M.I. Baskes, *Mater. Sci. Rep.* **9**, 251 (1993)
- M.I. Baskes, *Phys. Rev. Lett.* **59**, 2666 (1987)
- J.C. Slater, G.F. Koster, *Phys. Rev.* **94**, 1498 (1954)
- Th. Frauenheim, G. Seifert, M. Elsterner, Z. Hajnal, G. Jungnickel, D. Porezag, S. Suhai, R. Scholz, *Phys. Stat. Sol. B* **217**, 41 (2000)
- D.A. Papaconstantopoulos, M.J. Mehl, *J. Phys.: Condens. Matter* **15**, R413 (2003)
- R. Hoffman, *J. Chem. Phys.* **39**, 1397 (1963)
- R.J. Deeth, *Coord. Chem. Rev.* **212**, 11 (2001)
- R.J. Deeth, A. Anastasi, C. Diedrich, K. Randell, *Coord. Chem. Rev.* **253**, 795 (2009)
- A. Brown, B.J. Braams, K.M. Christoffel, Z. Jin, J.M. Bowman, *J. Chem. Phys.* **119**, 8790 (2003)
- Z. Xie, J.M. Bowman, *J. Chem. Theory Comput.* **6**, 26 (2010)
- B.J. Braams, J.M. Bowman, *Int. Rev. Phys. Chem.* **28**, 577 (2009)
- Y. Wang, B.C. Shepler, B.J. Braams, J.M. Bowman, *J. Chem. Phys.* **131**, 054511 (2009)
- X. Huang, B.J. Braams, J.M. Bowman, *J. Phys. Chem. A* **110**, 445 (2006)
- X. Huang, B.J. Braams, J.M. Bowman, *J. Chem. Phys.* **122**, 044308 (2005)
- A.R. Sharma, B.J. Braams, S. Carter, B.C. Shepler, J.M. Bowman, *J. Chem. Phys.* **130**, 174301 (2009)
- C.E. Rasmussen, C.K.I. Williams, *Gaussian Processes for Machine Learning* (MIT Press, 2006)
- C.M. Handley, G.I. Hawe, D.B. Kell, P.L.A. Popelier, *Phys. Chem. Chem. Phys.* **11**, 6365 (2009)
- M.J.L. Mills, P.L.A. Popelier, *Comput. Theor. Chem.* **975**, 42 (2011)
- M.J.L. Mills, P.L.A. Popelier, *Theor. Chem. Acc.* **131**, 1 (2012)
- M.J.L. Mills, G.I. Hawe, C.M. Handley, P.L.A. Popelier, *Phys. Chem. Chem. Phys.* **15**, 18249 (2013)
- T.J. Hughes, S.M. Kandathil, P.L.A. Popelier, *Spectrochim. Acta A Mol. Biomol. Spectrosc.*, in press (2013), DOI:10.1016/j.saa.2013.10.059
- A.P. Bartók, M.C. Payne, R. Kondor, G. Csányi, *Phys. Rev. B* **87**, 184115 (2013)
- A.P. Bartók, M.C. Payne, R. Kondor, G. Csányi, *Phys. Rev. Lett.* **104**, 136403 (2010)
- A.P. Bartók, M.J. Gillan, F.R. Manby, G. Csányi, *Phys. Rev. B* **88**, 054104 (2013)
- C. Lee, W. Yang, R.G. Parr, *Phys. Rev. B* **37**, 785 (1998)
- A.D. Becke, *Phys. Rev. A* **38**, 3098 (1998)
- M. Rupp, A. Tkatchenko, K.-R. Müller, O.A. von Lilienfeld, *Phys. Rev. Lett.* **108**, 058301 (2012)
- K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O.A. von Lilienfeld, A. Tkatchenko, K.R. Müller, *J. Chem. Theory Comput.* **9**, 3404 (2013)
- M. Rupp, M.R. Bauer, R. Wilcken, A. Lange, M. Reutlinger, F.M. Boeckler, G. Schneider, *PLoS Comput. Biol.* **10**, e1003400 (2014)
- P. Lancaster, K. Salkauskas, *Curve and Surface Fitting: An Introduction* (Academic Press, 1986)
- J. Ischtwan, M.A. Collins, *J. Chem. Phys.* **100**, 8080 (1994)
- M.J.T. Jordan, K.C. Thompson, M.A. Collins, *J. Chem. Phys.* **103**, 9669 (1995)
- M.J.T. Jordan, M.A. Collins, *J. Chem. Phys.* **104**, 4600 (1996)
- T. Wu, H.-J. Werner, U. Manthe, *Science* **306**, 2227 (2004)
- T. Wu, H.-J. Werner, U. Manthe, *J. Chem. Phys.* **124**, 164307 (2006)
- T. Takata, T. Taketsugu, K. Hiario, M.S. Gordon, *J. Chem. Phys.* **109**, 4281 (1998)
- C.R. Evenhuis, U. Manthe, *J. Chem. Phys.* **129**, 024104 (2008)
- C. Crespos, M.A. Collins, E. Pijper, G.J. Kroes, *Chem. Phys. Lett.* **376**, 566 (2003)
- C. Crespos, M.A. Collins, E. Pijper, G.J. Kroes, *J. Chem. Phys.* **120**, 2392 (2004)
- D.H. McLain, *Comput. J.* **17**, 318 (1974)
- T. Ishida, G.C. Schatz, *Chem. Phys. Lett.* **314**, 369 (1999)
- R. Dawes, D.L. Thompson, Y. Guo, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* **126**, 184108 (2007)
- R. Dawes, D.L. Thompson, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* **128**, 084107 (2008)
- G.G. Maisuradze, D.L. Thompson, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* **119**, 10002 (2003)

61. Y. Guo, A. Kawano, D.L. Thompson, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* **121**, 5091 (2004)
62. R. Dawes, A.F. Wagner, D.L. Thompson, *J. Phys. Chem. A* **113**, 4709 (2009)
63. J.P. Camden, R. Dawes, D.L. Thompson, *J. Phys. Chem. A* **113**, 4626 (2009)
64. R. Dawes, X.-G. Wang, T. Carrington, *J. Phys. Chem. A* **117**, 7612 (2013)
65. R. Dawes, X.-G. Wang, A.W. Jasper, T. Carrington, *J. Chem. Phys.* **133**, 134304 (2010)
66. J. Brown, X.-G. Wang, R. Dawes, T. Carrington, *J. Chem. Phys.* **136**, 134306 (2012)
67. G. Li, J. Hu, S.-W. Wang, P.G. Georgopoulos, J. Schoendorf, H. Rabitz, *J. Phys. Chem. A* **110**, 2474 (2006)
68. A. Kawano, I.V. Tokmakov, D.L. Thompson, A.F. Wagner, M. Minkoff, *J. Chem. Phys.* **124**, 054105 (2006)
69. C.M. Bishop, *Neural Networks for Pattern Recognition* (Oxford University Press, 1996)
70. S. Haykin, *Neural Networks and Learning Machines* (Pearson Education, 1986)
71. W. McCulloch, W. Pitts, *Bull. Math. Biophys.* **5**, 115 (1943)
72. J. Gasteiger, J. Zupan, *Angew. Chem. Int. Ed.* **32**, 503 (1993)
73. T.B. Blank, S.D. Brown, A.W. Calhoun, D.J. Doren, *J. Chem. Phys.* **103**, 4129 (1995)
74. E. Tafeit, W. Estelberger, R. Horejsi, R. Moeller, K. Oettl, K. Vrecko, G. Reibnegger, *J. Mol. Graphics* **14**, 12 (1996)
75. F.V. Prudente, P.H. Acioli, J.J. Soares Neto, *J. Chem. Phys.* **109**, 8801 (1998)
76. H. Gassner, M. Probst, A. Lauenstein, K. Hermansson, *J. Phys. Chem. A* **102**, 4596 (1998)
77. S. Lorenz, A. Groß, M. Scheffler, *Chem. Phys. Lett.* **395**, 210 (2004)
78. J. Behler, S. Lorenz, K. Reuter, *J. Chem. Phys.* **127**, 014705 (2007)
79. J. Ludwig, D.G. Vlachos, *J. Chem. Phys.* **127**, 154716 (2007)
80. C.M. Handley, P.L.A. Popelier, *J. Phys. Chem. A* **114**, 3371 (2010)
81. J. Behler, *Phys. Chem. Chem. Phys.* **13**, 17930 (2011)
82. J. Behler, *J. Phys.: Condens. Matter* **26**, 183001 (2014)
83. S. Haykin, *Neural Networks: A Comprehensive Foundation* (Macmillan College Publishing Company, 1994)
84. G. Cybenko, *Math. Control Signals Systems* **2**, 303 (1989)
85. K. Hornik, M. Stinchcombe, H. White, *Neural Netw.* **2**, 359 (1989)
86. D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Nature* **323**, 533 (1986)
87. T.B. Blank, S.D. Brown, *J. Chemometrics* **8**, 391 (1994)
88. A. Pukrittayakamee, M. Malshe, M. Hagan, L.M. Raff, R. Narulkar, S. Bukkapatnam, R. Komanduri, *J. Chem. Phys.* **130**, 134101 (2009)
89. N. Artrith, J. Behler, *Phys. Rev. B* **85**, 045439 (2012)
90. K.-H. Cho, K.-H. No, H.A. Scheraga, *J. Mol. Struct.* **641**, 77 (2002)
91. W.L. Jorgensen, J. Chandrasekhar, J.D. Madura, R.W. Impey, *J. Chem. Phys.* **79**, 962 (1983)
92. S. Manzhos, T. Carrington, *J. Chem. Phys.* **125**, 84109 (2006)
93. S. Manzhos, T. Carrington, *J. Chem. Phys.* **125**, 194105 (2006)
94. S. Manzhos, T. Carrington, *J. Chem. Phys.* **127**, 014103 (2007)
95. S. Manzhos, T. Carrington, *J. Chem. Phys.* **129**, 224104 (2008)
96. M. Malshe, R. Narulkar, L.M. Raff, M. Hagan, S. Bukkapatnam, P.M. Agrawal, R. Komanduri, *J. Chem. Phys.* **130**, 184101 (2009)
97. S. Hobday, R. Smith, J. BelBruno, *Modelling Simul. Mater. Sci. Eng.* **7**, 397 (1999)
98. S. Hobday, R. Smith, J. BelBruno, *Nucl. Instrum. Methods Phys. Res. B* **153**, 247 (1999)
99. A. Bhoola, S.D. Kenny, R. Smith, *Nucl. Instrum. Methods Phys. Res. B* **255**, 1 (2007)
100. E. Sanville, A. Bhoola, R. Smith, S.D. Kenny, *J. Phys.: Condens. Matter* **20**, 285219 (2008)
101. J. Behler, M. Parrinello, *Phys. Rev. Lett.* **98**, 146401 (2007)
102. J. Behler, *J. Chem. Phys.* **134**, 074106 (2011)
103. N. Artrith, T. Morawietz, J. Behler, *Phys. Rev. B* **83**, 153101 (2011)
104. T. Morawietz, V. Sharma, J. Behler, *J. Chem. Phys.* **136**, 064103 (2012)
105. J. Behler, R. Martoňák, D. Donadio, M. Parrinello, *Phys. Rev. Lett.* **100**, 185501 (2008)
106. R.Z. Khaliullin, H. Eshet, T.D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **81**, 100103 (2010)
107. H. Eshet, R.Z. Khaliullin, T.D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **81**, 184107 (2010)
108. N. Artrith, J. Behler, *Phys. Rev. B* **85**, 045439 (2012)
109. G.C. Sosso, G. Miceli, S. Caravati, J. Behler, M. Bernasconi, *Phys. Rev. B* **85**, 174103 (2012)
110. N. Artrith, B. Hiller, J. Behler, *Phys. Stat. Sol. B* **250**, 1191 (2013)
111. T. Morawietz, J. Behler, *J. Phys. Chem. A* **117**, 7356 (2013)
112. T. Morawietz, J. Behler, *Z. Phys. Chem.* **227**, 1559 (2013)
113. S. Houlding, S.Y. Liem, P.L.A. Popelier, *Int. J. Quant. Chem.* **107**, 2817 (2007)
114. M.G. Darley, C.M. Handley, P.L.A. Popelier, *J. Chem. Theor. Comput.* **4**, 1435 (2008)
115. C.M. Handley, P.L.A. Popelier, *J. Chem. Theor. Chem.* **5**, 1474 (2009)
116. P.L.A. Popelier, *Atoms in Molecules; An Introduction* (Pearson Education, 2000)
117. P.L.A. Popelier, F.M. Aicken, *ChemPhysChem* **4**, 824 (2003)
118. P.L.A. Popelier, A.G. Brémond, *Int. J. Quant. Chem.* **109**, 2542 (2009)
119. J. Li, B. Jiang, H. Guo, *J. Chem. Phys.* **139**, 204103 (2013)
120. R. Fournier, O. Slava, *J. Chem. Phys.* **139**, 234110 (2013)
121. P. Geiger, C. Dellago, *J. Chem. Phys.* **139**, 164105 (2013)
122. P.J. Haley, D. Soloway, in *International Joint Conference on Neural Networks, IJCNN, 1992*, Vol. 4, pp. 25–30
123. A.G. Wilson, E. Gilboa, A. Nehorai, J.P. Cunningham, [arXiv:1310.5288v3](https://arxiv.org/abs/1310.5288v3) (2013)
124. V.N. Vapnik, *The Nature of Statistical Learning Theory* (Springer-Verlag, 1995)
125. C. Cortes, V. Vapnik, *Mach. Learn.* **20**, 273 (1995)
126. Z.R. Yang, *Brief. Bioinform.* **5**, 328 (2004)

127. R.M. Balabin, E.I. Lomakina, *Phys. Chem. Chem. Phys.* **13**, 11710 (2011)
128. W. Chu, S.S. Keerthi, C.J. Ong, *IEEE Trans. Neural Networks* **15**, 29 (2004)
129. A. Vitek, M. Stachon, P. Kromer, V. Snael, in *International Conference on Intelligent Networking and Collaborative Systems (INCoS), 2013*, pp. 121–126
130. J.R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection* (MIT Press, 1992)
131. D.E. Makarov, H. Metiu, *J. Chem. Phys.* **108**, 590 (1998)
132. M.A. Bellucci, D.F. Coker, *J. Chem. Phys.* **135**, 044115 (2011)
133. M.W. Brown, A.P. Thompson, P.A. Schultz, *J. Chem. Phys.* **132**, 024108 (2010)
134. E. Zitzler, L. Thiele, *IEEE Trans. Evol. Comput.* **3**, 257 (1999)
135. D.E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning* (Addison-Wesley, 1998)
136. B. Hartke, *Struct. Bond.* **110**, 33 (2004)
137. A.R. Oganov, C.W. Glass, *J. Chem. Phys.* **124**, 244704 (2006)
138. F. Koskowsky, B. Hartke, *J. Comput. Chem.* **26**, 1169 (2005)
139. W. Leo Meerts, M. Schmitt, *Int. Rev. Phys. Chem.* **25**, 353 (2006)
140. M.H. Hennessy, A.M. Kelley, *Phys. Chem. Chem. Phys.* **6**, 1085 (2004)
141. J.M.C. Marques, F.V. Prudente, F.B. Pereira, M.M. Almeida, M.M. Maniero, C.E. Fellows, *J. Phys. B* **41**, 085103 (2008)
142. W.F. Da Cunha, L.F. Roncaratti, R. Gargano, G.M.E. Silva, *Int. J. Quant. Chem.* **106**, 2650 (2006)
143. L.F. Roncaratti, R. Gargano, G.M.E. Silva, *J. Mol. Struct. (Theochem)* **769**, 47 (2006)
144. Y.G. Xu, G.R. Liu, *J. Micromech. Microeng.* **13**, 254 (2003)
145. G.L.W. Hart, V. Blum, M.J. Walorski, A. Zunger, *Nat. Mater.* **4**, 391 (2005)
146. V. Blum, G.L.W. Hart, M.J. Walorski, A. Zunger, *Phys. Rev. B* **72**, 165113 (2005)
147. P. Pahari, S. Chaturvedi, *J. Mol. Model.* **18**, 1049 (2012)
148. H.R. Larsson, A.C.T. van Duin, B.J. Hartke, *Comput. Chem.* **34**, 2178 (2013)
149. C.M. Handley, R.J. Deeth, *J. Chem. Theor. Comput.* **8**, 194 (2012)