

PAPER • OPEN ACCESS

Improved description of atomic environments using low-cost polynomial functions with compact support

To cite this article: Martin P Bircher *et al* 2021 *Mach. Learn.: Sci. Technol.* **2** 035026

View the [article online](#) for updates and enhancements.

You may also like

- [Materials can be strengthened by nanoscale stacking faults](#)
J. Wang, Y. G. Shen, F. Song et al.
- [Homoepitaxial growth and investigation of stacking faults of 4H-SiC C-face epitaxial layers with a 1° off-angle](#)
Keiko Masumoto, Hirokuni Asamizu, Kentaro Tamura et al.
- [A Closer Look at the Role of Nanometer Scale Solute-Rich Stacking Faults in the Localized Corrosion of a Magnesium Alloy GZ31K](#)
X. Zhang, S. K. Kairy, J. Dai et al.



PAPER

OPEN ACCESS

RECEIVED
22 January 2021REVISED
28 March 2021ACCEPTED FOR PUBLICATION
14 April 2021PUBLISHED
16 June 2021

Original Content from
this work may be used
under the terms of the
[Creative Commons
Attribution 4.0 licence](#).

Any further distribution
of this work must
maintain attribution to
the author(s) and the title
of the work, journal
citation and DOI.



Improved description of atomic environments using low-cost polynomial functions with compact support

Martin P Bircher^{1,*} , Andreas Singraber^{1,2} and Christoph Dellago¹ ¹ Faculty of Physics, University of Vienna, Boltzmannngasse 5, Wien A-1090, Austria² Institute for Theoretical Physics, TU Wien—Vienna University of Technology, Wiedner Hauptstraße 8-10, Wien A-1040, Austria

* Author to whom any correspondence should be addressed.

E-mail: martin.bircher@univie.ac.at**Keywords:** computational chemistry, computational physics, descriptors, symmetry functions, high-dimensional neural network potential (HDNNP), condensed matterSupplementary material for this article is available [online](#)

Abstract

The prediction of chemical properties using machine learning techniques calls for a set of appropriate descriptors that accurately describe atomic and, on a larger scale, molecular environments. A mapping of conformational information on a space spanned by atom-centred symmetry functions (SF) has become a standard technique for energy and force predictions using high-dimensional neural network potentials (HDNNP). An appropriate choice of SFs is particularly crucial for accurate force predictions. Established atom-centred SFs, however, are limited in their flexibility, since their functional form restricts the angular domain that can be sampled without introducing problematic derivative discontinuities. Here, we introduce a class of atom-centred SFs based on polynomials with compact support called polynomial symmetry functions (PSF), which enable a free choice of both, the angular and the radial domain covered. We demonstrate that the accuracy of PSFs is either on par or considerably better than that of conventional, atom-centred SFs. In particular, a generic set of PSFs with an intuitive choice of the angular domain inspired by organic chemistry considerably improves prediction accuracy for organic molecules in the gaseous and liquid phase, with reductions in force prediction errors over a test set approaching 50% for certain systems. Contrary to established atom-centred SFs, computation of PSF does not involve any exponentials, and their intrinsic compact support supersedes use of separate cutoff functions, facilitating the choice of their free parameters. Most importantly, the number of floating point operations required to compute polynomial SFs introduced here is considerably lower than that of other state-of-the-art SFs, enabling their efficient implementation without the need of highly optimised code structures or caching, with speedups with respect to other state-of-the-art SFs reaching a factor of 4.5 to 5. This low-effort performance benefit substantially simplifies their use in new programs and emerging platforms such as graphical processing units. Overall, polynomial SFs with compact support improve accuracy of both, energy and force predictions with HDNNPs while enabling significant speedups compared to their well-established counterparts.

1. Introduction

In the past decade, computational chemistry has seen a tremendous increase in use of machine learning (ML) techniques to overcome the length- and timescale problem burdening *first principles* methods [1–4]. As such, ML techniques have seen particularly beneficial use in molecular dynamics (MD) simulations in the condensed phase. By providing highly elaborate fitting mechanisms, ML allows for accurate interpolation between training points on the potential energy surface (PES) generated from a computationally more expensive reference method. This enables, e.g., MD simulations of water to be performed at the nano- rather

than the picosecond scale with forces and energies of *first principles* accuracy [5, 6]. While still more expensive than classical force fields, ML techniques can incorporate information from the underlying PES well outside of local minima, making the simulation of chemical reactions possible. In contrast to reactive force-fields, no assumptions on the functional form of the interactions and therefore, the underlying PES, have to be made [7, 8]. Instead, the fit is carried out mostly in a black-box manner, with one of the most prominent adjustable input parameters being a set of functions that uniquely describe the environment of every atom in a manner that is invariant both to permutation, rotation and translation [4, 9, 10].

Machine learning techniques commonly used to interpolate between *first principles* data include kernel-based methods [11–14] and neural network potentials (NNP) [15, 16]. A prominent framework of the latter is due to Behler and Parrinello [17, 18], who advocated the use of high-dimensional neural network potentials (HDNNP) in combination with atom-centred input functions. In the context of HDNNPs, these functions are commonly referred to as symmetry functions (SF) [19]. Ideally, these functions provide a unique description of the atomic environment of every structure in the training set. Several functional forms have been proposed [20, 21], which commonly include a radial and angular part multiplied by a cutoff function that ensures that the former are naught outside of predefined bounds. SFs are therefore commonly a product of up to three different types of functions, and parametrising them properly for a given chemical system can become non-trivial. An intuitive construction of SFs can be based on structural features of the system at hand. For instance, sufficient spatial resolution could be obtained by taking into account all relevant peaks in the radial distribution function of the system and by covering all chemically relevant angles spanned by an atom and two of its neighbours. Note that in the context of HDNNPs, the term neighbour may refer to any atom in the vicinity of another atom. A neighbouring atom may hence belong either to the same or a different molecule, and it may be chemically connected to its neighbour by some kind of bond or not at all. The width and centre of a SF should then reflect fluctuations of the quantities in the training data that a given SF is expected to capture. Such a strategy has successfully been used, e.g. for the construction of a neural network of water and ice, including different ice phases [5, 22, 23].

However, whereas such a non-automated approach can yield SF sets much smaller than those generated using automated procedures [21, 24], there are some pitfalls associated to an intuitive approach: while the position of the maxima of radial functions can generally be freely chosen within the cutoff radius, this is not necessarily the case for angular terms, which are commonly based on a cosine and which are therefore restricted to peak at either 0 or π . Introducing a simple phase shift in the cosine has been shown to work well for energy evaluations [25], but it would be bound to fail for force predictions, since any shift in a cosine will inevitably introduce derivative discontinuities at 0 and π : The evaluation of angles is restricted to a domain of $[0, \pi]$, imposing symmetries in the angular functions of the form $f(\vartheta) = f(-\vartheta)$ and $f(\pi + \vartheta) = f(\pi - \vartheta)$. Phase shifts break this symmetry and introduce non-zero derivatives at $f(0)$ and $f(\pi)$. Since derivatives of the angular functions are used to compute forces (*vide infra*), this results in non-unique descriptions at $\vartheta = 0$ and $\vartheta = \pi$.

In order to account for equilibrium angles that are not centred on either 0 or π and which only fluctuate over a few degrees — e.g. in a double bond — one therefore uses linear combinations of angular SFs which are centred on either 0 or π , increasing the overall number of SFs used. The presence of cutoff functions also obfuscates the choice of input SF parameters: If the SFs are constructed from structural properties such as the radial distribution function of a liquid, the position of the radial peaks is not easily deduced from the input parameters, since the product of radial term and cutoff function may exhibit a shifted maximum and a more rapid decay compared to the Gaussian function alone. In particular, if the Gaussian is not centred on the origin, this may lead to a SF that is not symmetric with respect to its peak. Therefore, it is usually necessary to plot every single SF in order to monitor the correspondence between the input parameters and the final system setup. Hence, parametrising SFs based on structural properties can be a cumbersome procedure. While universal rules to generate input SFs have been proposed [21, 24], they often increase the number of SFs needed when compared to an optimal, tailored set of parameters. This can hamper performance.

When using high-quality first principles methods to generate the training set, the first principle calculations usually represent the computational bottleneck. However, both the training of the neural network itself as well as its use in productive calculations carry a certain overhead that is typically larger than that of classical force fields. The computational cost of the training is in part associated to the number of SFs used, as well as their computational complexity. While it is possible to store the values of all SFs during training [26], making their calculation necessary only once at the very beginning of the training procedure, this is evidently impossible when the NNP is used to predict unknown structures e.g. during a MD run. Factors that contribute to the overhead of SF evaluation are two-fold: The most popular choice for the radial term are Gaussian functions. However, exponentials are comparably expensive to compute. Furthermore, the calculation of a cutoff function (often cosines or hyperbolic tangents) introduces an additional overhead with respect to bare radial functions. To this end, optimised cutoff functions based on polynomials have

already been proposed. Still, the disadvantages of the evaluation of exponentials for the radial terms, their possibly becoming asymmetric due to multiplication with a cutoff function and the impracticability of introducing phase shifts in the angular components remain. While strategies to reduce the large number of floating point operations during SF evaluation have been developed, e.g. by grouping and caching [26, 27], this involves intermediate storage of quantities, loop breaks and many `if`-statements, rendering their implementation cumbersome, reducing code readability and making performance benefits both problem- and architecture dependent.

It would therefore be desirable to have a framework at hand where the number of floating point operations in SF evaluations is substantially reduced, where the SF remain symmetric with respect to their maxima, where the input parameters directly reflect the shape of the resulting function, and where the maximum of the angular component can be adjusted to reflect equilibrium angles. This would alleviate the need to plot every SF during construction, making the resulting procedure more efficient and less error-prone, greatly simplifying the choice of parameters based on structural properties of the system at hand. Most importantly, a reduction of floating point operations supersedes the need of complex code optimisation and renders the efficient implementation of SFs architecture-independent.

In the following, we will demonstrate that further simplifications to existing SFs are possible by replacing the product-based ansatz of radial, angular and cutoff function by a product of polynomials with compact support, making the use of cutoff functions obsolete. Such polynomial symmetry functions (PSFs) have the advantage of allowing for angular terms to be centred anywhere within $\vartheta \in [0, \pi]$ without introducing derivative discontinuities at $\vartheta = 0$ and $\vartheta = \pi$. The latter is of particular importance for force evaluations, as discontinuities in SF derivatives will directly impact the forces computed. The evaluation of these PSFs does not involve exponentiation. Instead, the most demanding low-level operation becomes the calculation of an arccosine when computing angular terms. The use of one simple functional form for both angular and radial terms considerably reduces the number of floating point operations associated to the evaluation of a single SF. This does not only result in significant simplifications in the underlying code, but comes at the benefit of speedups of up to a factor of about 4.5 to 5, while their performance with respect to highly optimised cache and grouping based SF implementations still reaches a factor of almost 2 without the need of further algorithmic optimisations.

This text is organised as follows: first, we briefly introduce the concept of HDNNPs and describe the mathematical form of commonly used SFs first proposed by Behler and Parrinello [4, 17]. We then introduce our new PSFs. We will go on to demonstrate that use of PSFs can improve accuracy of NNPs when compared to conventional SFs. To this end, both dynamic and static properties will be compared for a water and copper sulphide model system. By training a HDNNP for the rotation of the amino group of the organic dye DMABN [28, 29], we will show that the angular flexibility of PSFs simplifies the choice of angular parameters compared to Behler–Parrinello type symmetry functions (BPSF). HDNNPs for liquid ethyl benzene and anisole that were trained using both PSF and BPSF will further support this perspective. Finally, we will show that use of PSF in conjunction with recently proposed weighted atom-centred symmetry functions (wACSF) [21] can further improve performance of HDNNPs for predicting enthalpies of formation of the QM9 [30] database. We will conclude the discussion by comparing execution times for the evaluation of the NNPs between BPSFs and PSFs, showing that the execution time per SF is substantially reduced for our polynomial ansatz.

2. Symmetry functions for high-dimensional neural network potentials

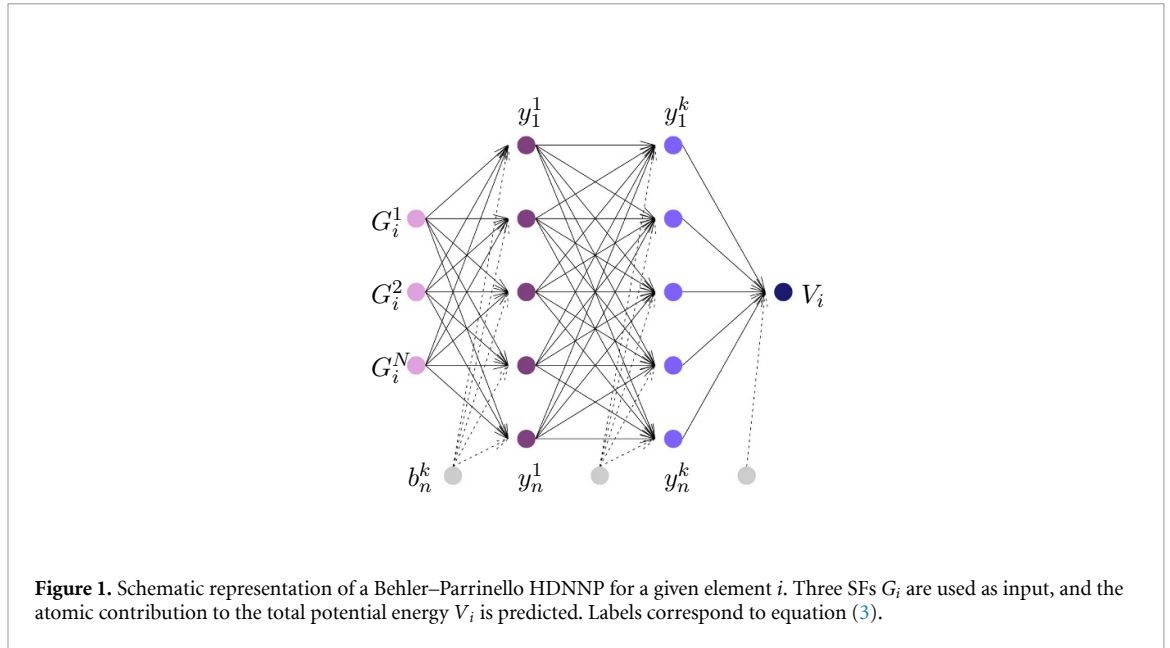
2.1. Neural networks in a nutshell

First applications of NNPs in computational chemistry date back as far as 1995 [15]. A considerable advance in the prediction of molecular forces and energies was made in 2007, when Behler and Parrinello [17] proposed to use HDNNP in combination with a fictitious decomposition of the potential energy V from N individual atoms:

$$V = \sum_i^N V_i(\mathbf{G}_i). \quad (1)$$

The atomic contribution V_i depends on the chemical environment of atom i , which is described by a tuple of N_j input values $\mathbf{G}_i = \{G_1, \dots, G_j\}$. The forces on the k th atom follow from the gradient:

$$-\nabla_k V = -\sum_i^N \sum_j^{N_j} \frac{\partial V_i}{\partial G_{i,j}} \nabla_k G_{i,j}. \quad (2)$$



The G_i are used as the input layer of a feed-forward neural network (figure 1) that yields the scalar V_i . The network itself is constituted by multiple layers of nodes, which are connected to each other by virtue of weights a_{nm} . Starting from a set of input values (G_i), the data is propagated to node n of hidden layer k as follows:

$$y_n^k = f_a \left(b_n^k + \sum_m^{M_l} a_{nm}^{lk} y_m^l \right), \quad (3)$$

where $l = k - 1$, y_n^k represents the value of the node and f_a is a non-linear activation function. b_n^k denotes the bias associated to y_n^k and M_l is the number of neurons in layer l . One independent neural network per element is used. Together with the biases b_n^k , the weights a_{nm} are the fitting parameters of the HDNNP.

The atomic environment descriptors G_i are commonly called SFs. In order to ensure permutational symmetry, the SFs are identical for all atoms of the same element. However, in order to account for varying chemical environments (bond lengths, van-der-Waals radii, relevant bond angles), they can differ between different elements. An appropriate choice of SFs is crucial in order to reliably discriminate between unique points on the PES.

2.2. Behler–Parrinello symmetry functions

In the case of Behler–Parrinello HDNNPs, the input layer consists of scalars obtained from a predetermined set of atom-centred SFs. The choice of SFs should discriminate between all relevant structural features, such that every distinct point on the PES is described by a unique tuple of SF values. In practice, this is achieved by combining a set of spherically symmetric and angle-dependent functions that have to be parametrised appropriately. Some examples are given in figure 2. For the radial part, Behler and Parrinello suggested:

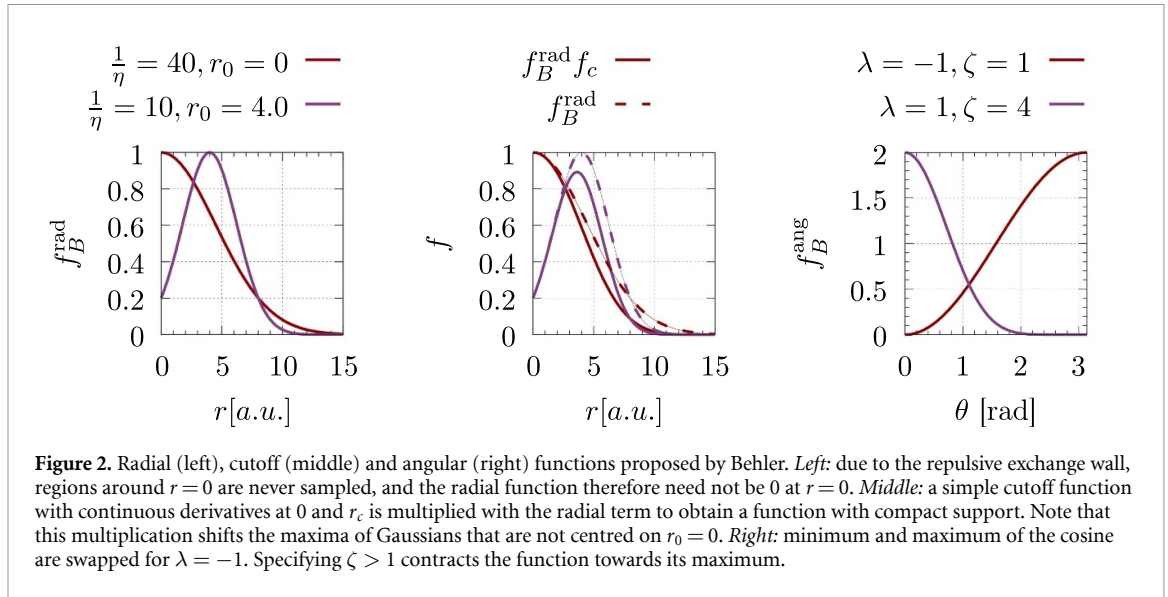
$$f_B^{\text{rad}}(r, r_s) = e^{-\eta(r-r_s)^2}, \quad (4)$$

where r is an interatomic distance and r_s its reference point. An angular dependency can be introduced by a function of the form:

$$f_B^{\text{ang}}(\vartheta, \lambda, \zeta) = (1 + \lambda \cos(\vartheta))^\zeta, \quad (5)$$

where ϑ is the angle formed by any three atoms, bound by $[0, \pi]$. This function appropriately has $\partial f_B^{\text{ang}} / \partial \vartheta|_{\vartheta=0} = \partial f_B^{\text{ang}} / \partial \vartheta|_{\vartheta=\pi} = 0$. For $\lambda = 1$, maximum and minimum of the function lie at 0 and π , respectively; for $\lambda = -1$ the inverse holds. Choosing $\zeta > 1$ allows to contract the function, resulting in a more rapid decay. Values of $\zeta < 1$ are not possible, since this would result in non-zero derivatives at the boundaries.

While the form of $f_B^{\text{ang}}(\vartheta, \lambda, \zeta)$ guarantees zero derivatives at the boundaries for appropriate choices of λ and ζ , the exponential in $f_B^{\text{rad}}(r, r_c)$ is finite everywhere. Therefore, a cutoff function with cutoff radius r_c is



introduced, which ensures that the radial term and its derivatives are zero for all $r > r_c$. Common choices of cutoff function $f_c(r, r_c)$ include:

$$f_c(r, r_c) = \begin{cases} \frac{1}{2} \left[\cos\left(\frac{\pi r}{r_c}\right) + 1 \right] & \text{for } r \leq r_c \\ 0 & \text{for } r > r_c \end{cases}, \quad (6)$$

or

$$f_c(r, r_c) = \begin{cases} \tanh^3\left(1 - \frac{r}{r_c}\right) & \text{for } r \leq r_c \\ 0 & \text{for } r > r_c \end{cases}. \quad (7)$$

With this choice of functions, the radial SF G_i centred on an atom i is given by a sum over all its neighbours j that are within r_c of i :

$$G_i^{\text{rad}}(r_c, r_s) = \sum_{j \neq i} f_B^{\text{rad}}(r_{ij}, r_s) f_c(r_{ij}, r_c). \quad (8)$$

Note that such a product may have a width and a maximum different from f_B^{rad} alone, see also figure 2. The angular SF is constructed from a product of angular and radial terms, with the sum encompassing all neighbours j and $k \neq j$. Depending on the radial term, two types of angular SFs can be distinguished, namely narrow n ,

$$G_i^{\text{ang},n}(r_c, r_s, \lambda, \zeta) = 2^{(1-\zeta)} \sum_{j \neq i, k > j} f_B^{\text{rad}}(r_{ij}, r_s) f_B^{\text{rad}}(r_{ik}, r_s) f_B^{\text{rad}}(r_{jk}, r_s) \quad (9)$$

$$\times f_B^{\text{ang}}(\vartheta_{ijk}, \lambda, \zeta) f_c(r_{ij}, r_c) f_c(r_{ik}, r_c) f_c(r_{jk}, r_c), \quad (10)$$

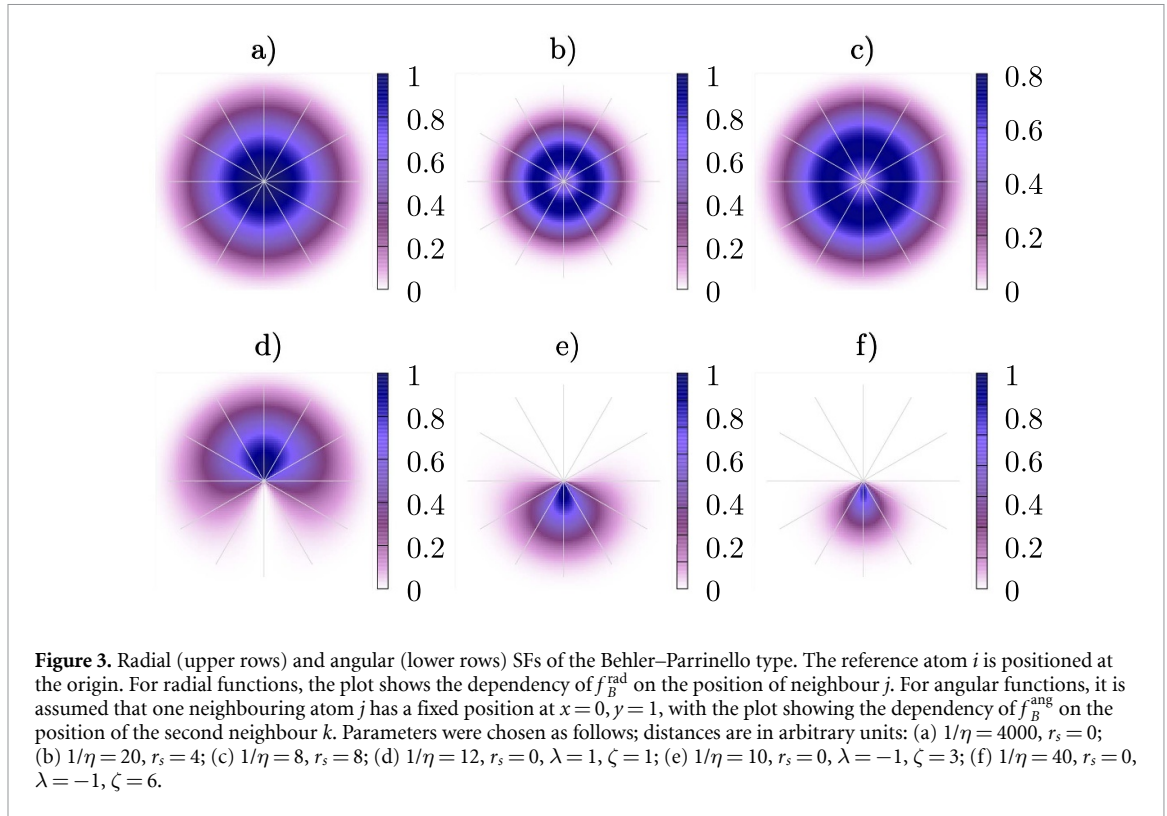
which also constrains the interatomic distance between atoms j and k , and the wide form w :

$$G_i^{\text{ang},w}(r_c, r_s, \lambda, \zeta) = 2^{(1-\zeta)} \sum_{j \neq i, k > j} f_B^{\text{rad}}(r_{ij}, r_s) f_B^{\text{rad}}(r_{ik}, r_s) f_B^{\text{ang}}(\vartheta_{ijk}, \lambda, \zeta) f_c(r_{ij}, r_c) f_c(r_{ik}, r_c), \quad (11)$$

the evaluation of which is more computationally expedient due to the lack of a radial term for jk . These SFs have successfully been applied in a wide variety of contexts. Note that the distribution of SF values at $\lambda = -1$ is not symmetric with respect to $\lambda = +1$ due the presence of a radial function [21]. Figure 3 displays the dependency of BPSFs on the position of atomic neighbour(s).

Recently, Singraber *et al* have introduced a simple expression for f_c based on polynomials [27]:

$$f_c^{\text{poly}}(r, r_c) = \begin{cases} f^{\text{poly}2}\left(\frac{r}{r_c}\right) & \text{for } 0 \leq r \leq r_c \\ 0 & \text{for } r > r_c, \end{cases} \quad (12)$$



with the polynomial:

$$f^{\text{poly}2}(x) = x^3(x(15 - 6x) - 10) + 1, \quad (13)$$

with $f^{\text{poly}2}(0) = 1$ and $f^{\text{poly}2}(1) = 0$. At $x = 0$ and $x = 1$, equation (13) has continuous derivatives up to second order, which ensures a continuous, smooth transition to outside of the compact set. Similar polynomials with continuous derivatives up to even higher order can be easily constructed [27], but their use does not result in any practical benefit [18]. The main advantage of polynomial cutoff functions as introduced in [27] lies in their expedient evaluation compared to expressions based on hyperbolic tangent or cosine, reducing the overhead due to the cutoff function. However, the computation of costly exponentials is still needed for all radial components. Hence, a grouping strategy was proposed in [27] which avoids multiple evaluations of the same exponential term, resulting in considerable speedups.

2.3. Polynomial symmetry functions with compact support

On the one hand, the summation over the arguments of G^{ang} still involves a substantial number of exponential functions that have to be explicitly evaluated even when a grouping strategy is used. On the other hand, the present choice of f^{ang} implies that, in order to specifically sample angles which are not centred close to either 0 or π , a combination of several values of λ and ζ is necessary, since introducing an angular shift $\vartheta = 0$ would inevitably violate the boundary condition $\partial f^{\text{ang}}/\partial \vartheta = 0$ at the bounds of the angular domain. In the following, we will attempt to overcome this issue by constructing a simple set of SFs based on polynomials with compact support which can be used to indiscriminately describe both angular and radial dependencies.

In order to do so, we generalise equation (12) to a generic form, expressed in terms of the boundaries x_{\min}, x_{\max} of the underlying domain. In particular, for our polynomial SFs f_{PSF} , we also allow for arguments $x < 0$:

$$f_{\text{PSF}}(x, x_0, \Delta x) = \begin{cases} f^{\text{poly}2}\left(\frac{x-x_0}{\Delta x}\right) & \text{for } x_0 \leq x \leq x_0 + \Delta x \\ f^{\text{poly}2}\left(\frac{x_0-x}{\Delta x}\right) & \text{for } x_0 - \Delta x \leq x < x_0 \\ 0 & \text{elsewhere,} \end{cases} \quad (14)$$

where $x_0 = \frac{1}{2}(x_{\max} + x_{\min})$ and $\Delta x = \frac{1}{2}(x_{\max} - x_{\min})$. The resulting function is symmetric with respect to x_0 and has continuous derivatives — up to an order determined by the construction of $f^{\text{poly}2}$ — on all \mathbb{R}^3 . It is possible to further influence the decay of the radial function by appropriately modifying its argument such

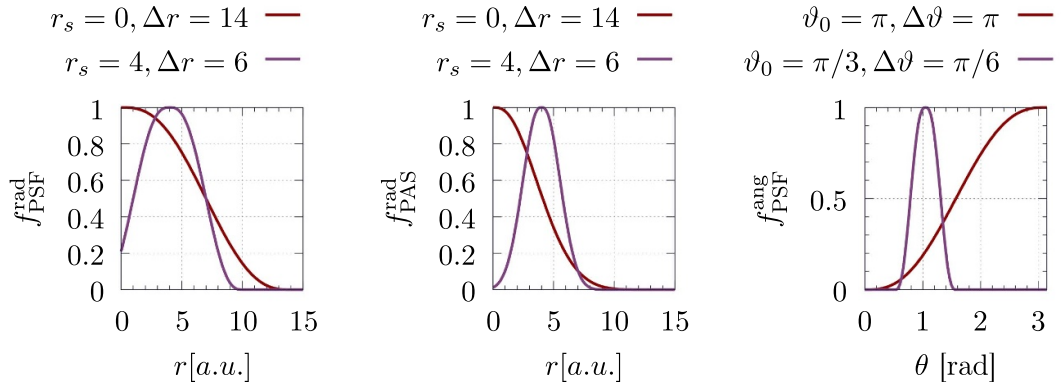


Figure 4. Radial PSF (left), radial PAS (middle) and angular (right) SFs with compact support proposed here. *Left:* radial function with point-symmetry around $0.5r_{\max}$. Note that no cutoff function is needed, which enables a precise choice of maxima and shifts. *Middle:* asymmetric radial function with longer tail, with a shape comparable to the product of a Gaussian and a cutoff function in figure 2. *Right:* angular polynomial SF. Note that the angular domain can be freely chosen and does not need to have maxima at 0 or π .

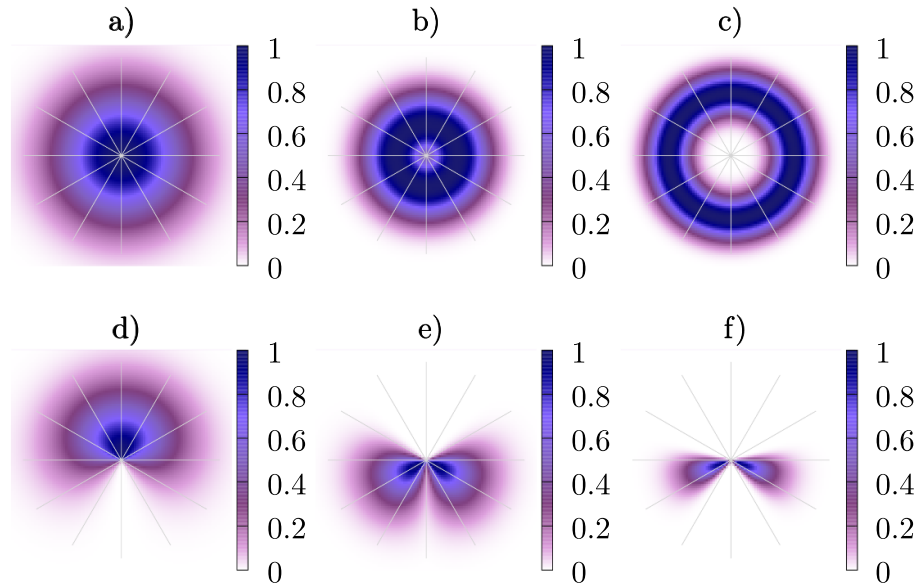


Figure 5. Radial (upper rows) and angular (lower rows) symmetry functions of the PAS type. The reference atom i is positioned at the origin. For radial functions, the plot shows the dependency of f_B^{rad} on the position of neighbour j . For angular functions, it is assumed that one neighbouring atom j has a fixed position at $x=0, y=1$, with the plot showing the dependency of f_B^{ang} on the position of the second neighbour k . Note that with PSF and PAS it becomes possible to cover specific angular ranges associated, e.g. to double bonds (60° angle, lower middle) or tetrahedra (109.1° , lower right). Parameters were chosen as follows; distances are in arbitrary units: (a) $\Delta r = 12, r_s = 0$; (b) $\Delta r = 6, r_s = 4$; (c) $\Delta r = 4, r_s = 8$; (d) $\Delta r = 12, r_s = 0, \Delta\vartheta = 90^\circ, \vartheta_0 = 0^\circ$; (e) $\Delta r = 10, r_s = 0, \Delta\vartheta = 40^\circ, \vartheta_0 = 120^\circ$; (f) $\Delta r = 8, r_s = 0, \Delta\vartheta = 20^\circ, \vartheta_0 = 109.1^\circ$.

that the SF decays more rapidly around its maximum while also approaching 0 more slowly, thus breaking the point symmetry around $f_{\text{PSF}} = 0.5$. We therefore define the asymmetric PAS as:

$$f_{\text{PAS}}(x, x_0, \Delta x) = \begin{cases} f^{\text{poly}2} \left(\frac{2(x-x_0)}{\Delta x} - \left(\frac{x-x_0}{\Delta x} \right)^2 \right) & \text{for } x_0 \leq x \leq x_0 + \Delta x \\ f^{\text{poly}2} \left(\frac{2(x_0-x)}{\Delta x} - \left(\frac{x_0-x}{\Delta x} \right)^2 \right) & \text{for } x_0 - \Delta x \leq x < x_0 \\ 0 & \text{elsewhere,} \end{cases} \quad (15)$$

which offers the same benefits as f_{PSF} for arguments $\in [0, 1]$. By construction, f_{PAS} resembles the shape of the product of a local, narrow Gaussian function with a cutoff function and can therefore be used to adapt existing BPSF-setups to polynomial form. The functional forms of f_{PAS} and f_{PSF} are depicted in figure 4.

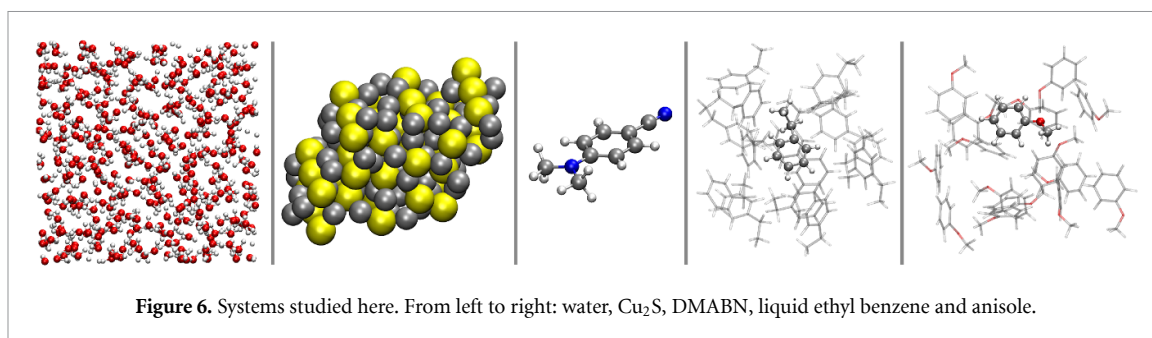


Figure 6. Systems studied here. From left to right: water, Cu₂S, DMABN, liquid ethyl benzene and anisole.

The corresponding radial SFs can now be constructed as:

$$G_i^{\text{rad/P}} = \sum_{j \neq i} f_P(r_{ij}, r_s, \Delta r), \quad (16)$$

with f_P either f_{PSF} or f_{PAS} . Their narrow angular counterparts read:

$$G_i^{\text{ang.n/P}} = \sum_{j \neq i, k > j} f_P(r_{ij}, r_s, \Delta r) f_P(r_{ik}, r_s, \Delta r) f_P(r_{jk}, r_s, \Delta r) f_{\text{PSF}}(\vartheta_{ijk}, \vartheta_0, \Delta \vartheta), \quad (17)$$

while the wide angular functions become:

$$G_i^{\text{ang.w/P}} = \sum_{j \neq i, k > j} f_P(r_{ij}, r_s, \Delta r) f_P(r_{ik}, r_s, \Delta r) f_{\text{PSF}}(\vartheta_{ijk}, \vartheta_0, \Delta \vartheta), \quad (18)$$

where the radial component f_P can again be described by both f_{PAF} or f_{PSF} . Both types of angular SF can be centred anywhere within $[0, \pi]$, provided that $\partial f_{\text{PSF}}/\partial \delta|_{\vartheta=0} = 0$ and $\partial f_{\text{PSF}}/\partial \delta|_{\vartheta=\pi} = 0$. This enables chemically intuitive choices of angular maxima, i.e. 60° for a double bond or $109, 4^\circ$ for a carbon-carbon single bond. Figure 5 shows the dependency of PAS-type SFs on the position of atomic neighbour(s).

Equations (14) and (15) do not involve any exponentials, and the most expensive arithmetic operation, an inverse cosine, has a modest computational footprint.

3. Computational methods

3.1. Training of the HDNNP

All HDNNPs were trained using the n2p2 package [26, 27, 31]. HDNNPs for the prediction of enthalpies of formation in the QM9 [30] database were obtained using the protocol of [21] with five-fold cross validation, but without dataset normalisation. For the remaining setups, 10 training runs per system are carried out with an independent seed for the random number generator each. If the value G_i of a given SF never exceeds 10^{-3} over the whole test set, the function is discarded before training. The choice of \mathbf{G} is detailed in the supporting information (available online at stacks.iop.org/MLST/2/035026/mmedia). Results discussed in this text refer to the best set of weights out of 10 independent runs (5 for the QM9 set), which are defined as the weights that yield the lowest force root mean-square deviation (RMSD) with respect to a test set constituted by 10% of training-set structures that were randomly removed before training. If several sets of weights yield force RMSDs that are identical to a few %, the set that displays the lowest relative error in energy predictions is used.

3.2. Training sets

The systems investigated here are depicted in figure 6. The training sets for water and copper sulphide correspond to those published in [5, 26], respectively. Forces and energies for an isolated DMABN molecule, liquid ethyl benzene and anisole (16 molecules in a periodic box) were obtained from high-temperature Car-Parrinello MD runs performed with the CPMD code [32] using the SCAN exchange-correlation functional [33]. The amino group of DMABN was rotated using a slowly growing harmonic restraint, giving rise to a set of non-equilibrium conformers. Details on the computational setup can be found in the supporting information.

3.3. Test sets

Test sets consists of 10% randomly chosen configurations that are removed from the training set before training and which are then used to validate the predictive power of the NNP.

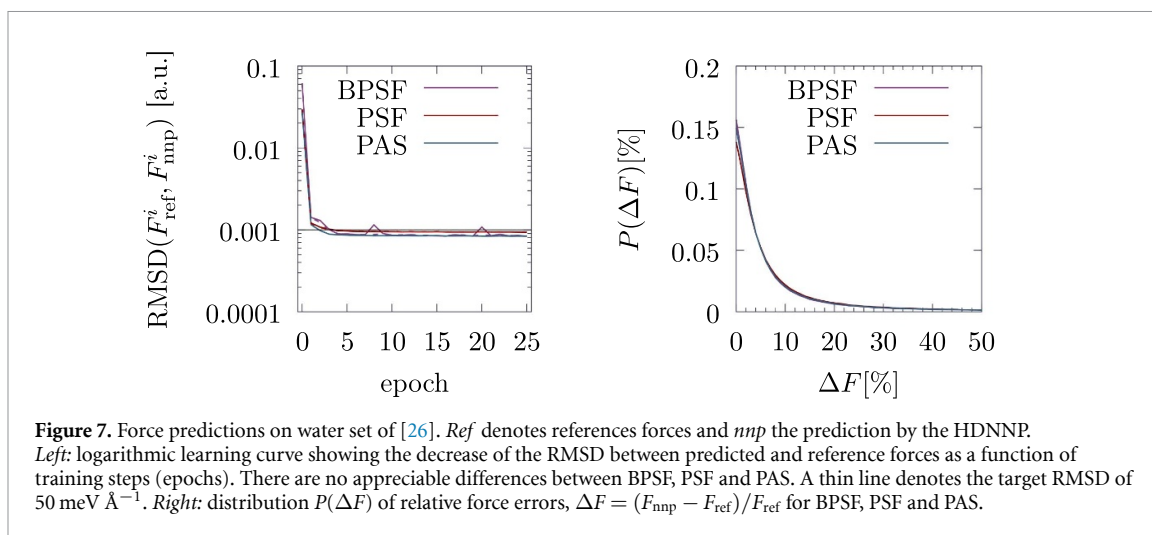


Figure 7. Force predictions on water set of [26]. *Ref* denotes reference forces and *nnp* the prediction by the HDNNP. *Left:* logarithmic learning curve showing the decrease of the RMSD between predicted and reference forces as a function of training steps (epochs). There are no appreciable differences between BPSF, PSF and PAS. A thin line denotes the target RMSD of $50 \text{ meV } \text{\AA}^{-1}$. *Right:* distribution $P(\Delta F)$ of relative force errors, $\Delta F = (F_{\text{nnp}} - F_{\text{ref}})/F_{\text{ref}}$ for BPSF, PSF and PAS.

3.4. Molecular dynamics using neural network potentials

All MD simulations using HDNNPs were carried out using the n2p2-interface [27, 31] to the LAMMPS code [34, 35]. System setups are detailed in the supporting information.

4. Results and discussion

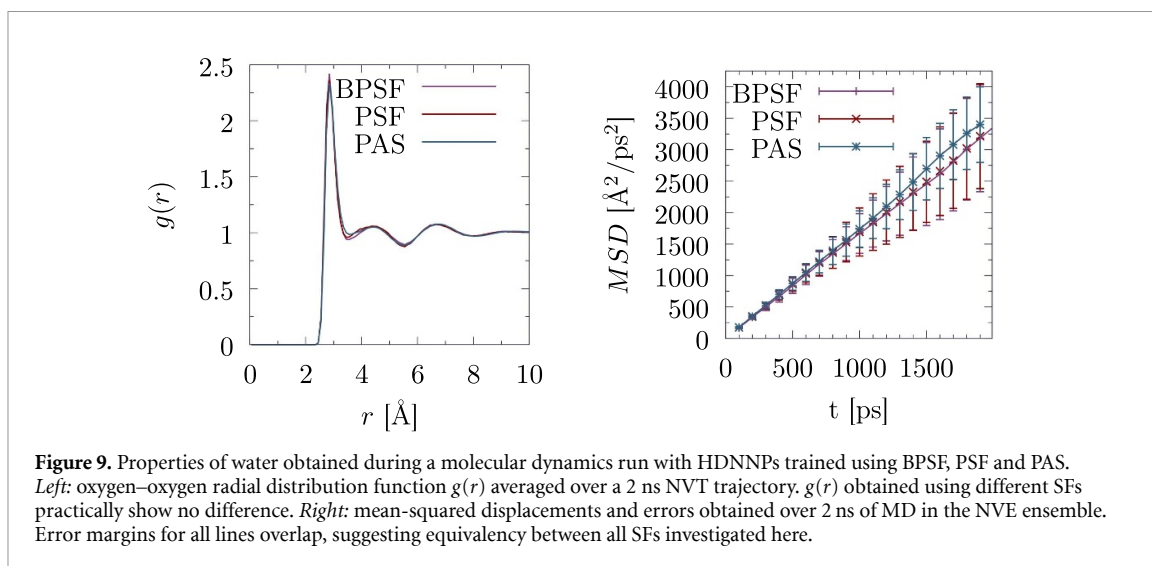
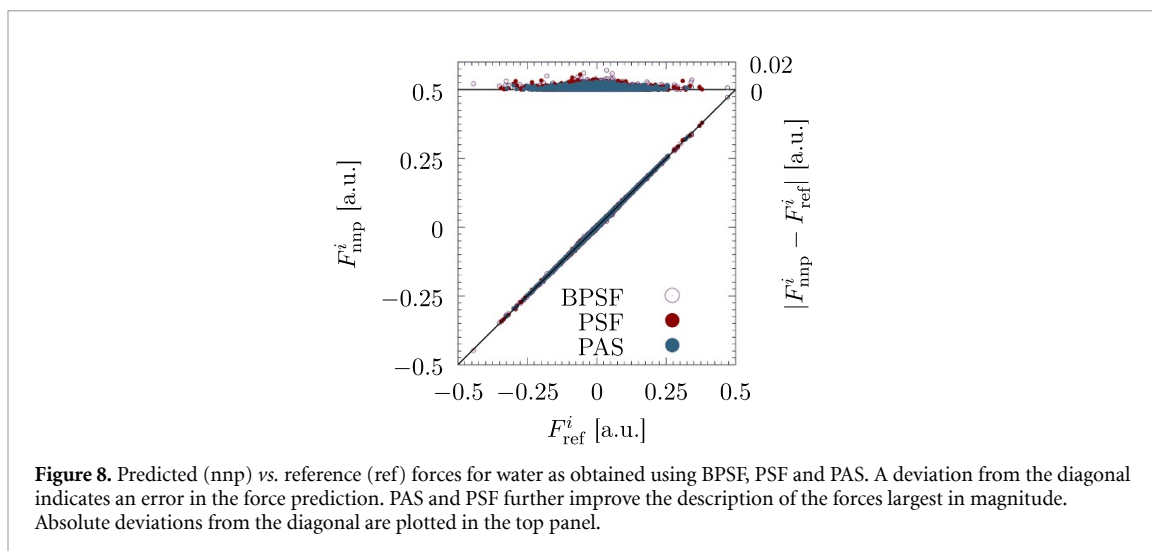
4.1. Water

We first assess the accuracy that can be attained with polynomial SFs by comparing the efficiency and accuracy of both training and productive simulations using identically set up HDNNPs. Two setups of polynomial SFs will be used: one, PSF, consisting entirely of symmetric functions of the form 14, and one mixed setup, PAS, including both symmetric and asymmetric radial components. We will compare against a reference setup using BPSFs, which has successfully been used in [26] to describe the PES of water and several ice phases. PSF and PAS were constructed by plotting the BPSFs and using their peaks and the full width at half maximum as rough indicators for the polynomial input. For sake of simplicity, the only symmetric radial terms (PSF-type) in the PAS set were those with maximum width; all others were chosen to be of the asymmetric type.

First, we compare the learning curves of both methods to assess the efficiency of the training procedure. Here, we define the learning curve as the RMSD between trained energy and forces and their reference as a function of the number of training steps (epochs). Curves for Behler–Parrinello type and polynomial SF setups are plotted in figure 7 and are virtually indistinguishable, suggesting that both types of symmetry functions can be trained with equal efficiency. RMSDs at the last training step are comparable between different methods, with the largest remaining force deviation being observed for the PSF setup. Conversely, the lowest force RMSD is observed for the PAS setup. BPSFs yield a training error intermediate between PAS and PSF data.

The same conclusion holds when analysing the distribution of both force and energy errors within the test set: Force RMSDs are $43.7 \text{ meV } \text{\AA}^{-1}$ for BPSFs, $48.5 \text{ meV } \text{\AA}^{-1}$ for PSF and $43.2 \text{ meV } \text{\AA}^{-1}$ for PAS, respectively. Energy RMSDs range from 1.17 meV/atom for BPSFs to 0.95 meV/atom for PAS. In practical applications, these values could be considered identical. In figure 7, we histogram both the relative force error for HDNNPs trained with Behler–Parrinello type and polynomial SFs. The resulting distributions for the test set compare well between PSF, PAS and BPSF: All relative errors show a rapid decay, indicating an excellent quality of the fit. PAS and BPSF show very similar relative error distributions, whereas the distribution of PSF skews very lightly towards the right; however, we will show that this is of no practical relevance. Figure 8 shows predicted forces against their reference values; ideally, all values should lie on a diagonal. Use of PAS and PSF improves the accuracy of the most extreme force values at either end of the graph, which lie closer to the diagonal than for BPSF. This indicates that, although force RMSDs differ only little between the different SF setups, qualitative improvements to conventional BPSFs are possible by use of a PSF or PAS set.

Figure 9 provides a practical comparison of the predictive power of the different HDNNPs. Both the radial distribution function (left panel) as well as the mean square displacement as a function of time, from which the diffusion coefficient can be derived, (right panel) are practically indistinguishable between BPSF, PAS and PSF. These results further support that polynomial SFs are a viable alternative to the commonly used Behler–Parrinello type functions. The excellent agreement between radial distribution functions and

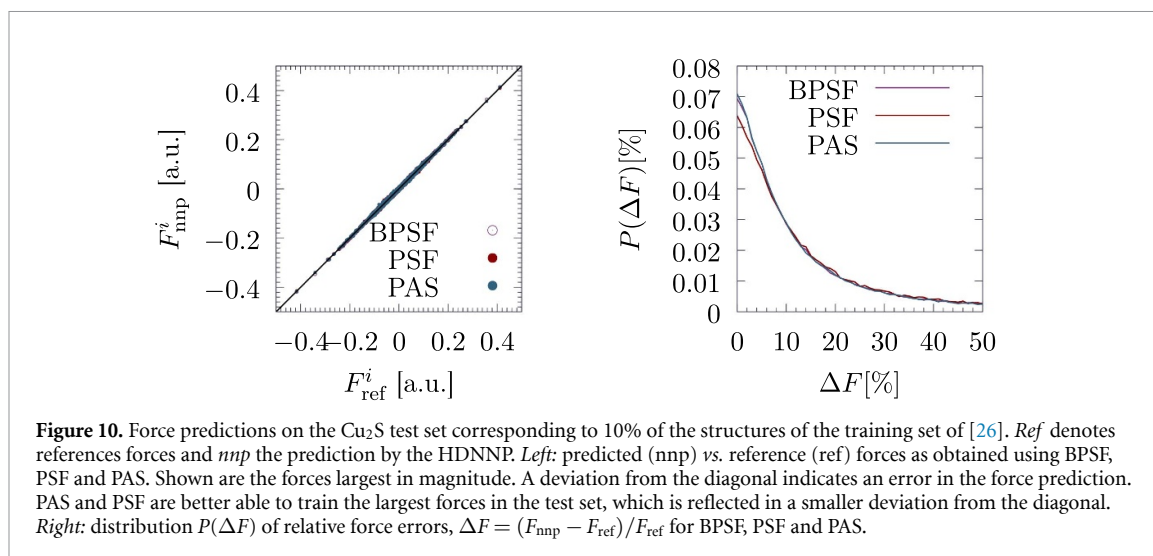


mean-squared displacements obtained from MD suggest that the small differences in force RMSD over the test set do not entail significant changes in the quality of the predictions made by the HDNNP. Notably, the mean-squared displacements (MSD) for PSF and BPSF are virtually identical.

4.2. Copper sulphide

All SF setups used for water were based on the optimised, hand-selected set of [5]. Such a selection by hand may not always be possible. In the following, we shall further validate the accuracy of PSF and PAS by comparing their performance to the accuracy of a semi-automatically generated set of BPSFs for copper sulphide. It has been shown in [26] that a HDNNP with BPSFs can be successfully used to observe structural phase transitions in this compound.

Figure 10 compares absolute and relative force errors obtained using BPSFs of [26] as well as our PSF and PAS. Force RMSDs are $51.9 \text{ meV } \text{Å}^{-1}$ for BPSF, $56.7 \text{ meV } \text{Å}^{-1}$ for PSF and $52.3 \text{ meV } \text{Å}^{-1}$ for PAS, which corresponds roughly to a 10% difference between PSF and BPSF. Again, the performance of BPSF and PAS is on par, whereas use of PSF leads to slightly larger errors, which is also reflected in the relative error distribution skewing lightly to the right. Overall, however, this effect is expected to remain negligible in practical applications. Deviations from the diagonal in the graph comparing prediction and reference forces (figure 10, right panel) do not exhibit visible differences, further indicating that the general performance of all SF types investigated here remains comparable and that the differences in force RMSDs observed here are not expected to influence the quality of the predictions in a significant manner. Energy RMSDs remain similar for all methods, spanning a range from 0.99 meV/atom for PAS to 1.06 meV/atom for BPSF.



4.3. Organic molecules: gas phase and liquids

In the following, we shall compare the performance of BPSFs generated according to [24] with our PSFs using an isolated (gas-phase) DMABN molecule and liquid ethyl benzene. We note that optimal SF setups could possibly be found for all SF types investigated here; however, since this is a time-intensive procedure, the main purpose of this comparison is to compare the out-of-the-box performance of some generic sets of SFs.

To this end, we introduce a simple generation scheme for radial SF parameters given a minimal and maximum width, Δ_{\min} and Δ_{\max} , and a maximum cutoff radius r_c . For N_0 SFs which are symmetric w.r.t. to the origin, we choose the width Δ_0^i parameters of the i th out of N_0 PSF or PAS as:

$$\Delta_0^0 = r_c \quad (19)$$

$$\Delta_0^i = \Delta_{\max} - \frac{(i-1)(\Delta_{\max} - \Delta_{\min})}{N_0 - 2} \quad ; \quad i \geq 1 \quad (20)$$

For a given number N_s of SFs which are centred at $r_s^i \neq 0$, we obtain shifts and widths Δ_s^i as:

$$\Delta_s^i = \frac{2r_c}{N_s + 1} \quad (21)$$

$$r_s^i = \frac{(i-1)\Delta_s^i}{2}. \quad (22)$$

This partitioning is reminiscent to the one introduced by Gastegger *et al* in [21]. Angles were chosen to reflect common structural features encountered in organic chemistry: linearity for triple bonds, 60° and 109.1° angles to describe double- and single carbon-carbon bonds, respectively. This resulted in the following angular functions: A wider set centred on $\vartheta_0^1 = 0^\circ$, $\vartheta_1^1 = 90^\circ$ and $\vartheta_3^1 = 180^\circ$ with a width of $\Delta\vartheta^{1,2,3} = 90^\circ$ each; and an additional set centred on $\vartheta_0^4 = 60^\circ$, $\vartheta_0^5 = 120^\circ$ and $\vartheta_0^6 = 109.1^\circ$, corresponding to a more specific sampling of environments typical of double- and single bonds, with a width of $\Delta\vartheta^{4,5,6} = 60^\circ$ each. For sake of comparison, the cutoff function distance for BPSFs and the maximum extent of PAS and PSF were set to 15 a.u.

Structures in the DMABN-set cover regions of the PES thermally accessible at 1320 K, including a forced rotation around the amino group along a non-equilibrium path. This results in many forces with large magnitude which are well outside of the equilibrium regime in which simple harmonic potentials can also yield good approximations. The DMABN set therefore serves as an assessment for the transferability of our polynomial SFs to structures which are considerably off-equilibrium.

Force RMSDs on the test set amount to $63.2 \text{ meV } \text{\AA}^{-1}$ for BPSF, $56.6 \text{ meV } \text{\AA}^{-1}$ for PSF and $54.0 \text{ meV } \text{\AA}^{-1}$ for PAS. Note that this amounts to a difference exceeding 15% between worst (BPSF) and best (PAS) set. However, the distribution of relative errors remain comparable between all SFs, as evidenced by the force histogram in the right panel of figure 11. Energy RMSDs range from 0.64 meV/atom for PSF to 0.72 meV/atom for BPSF, with the overall error remaining low for all types of functions. The force

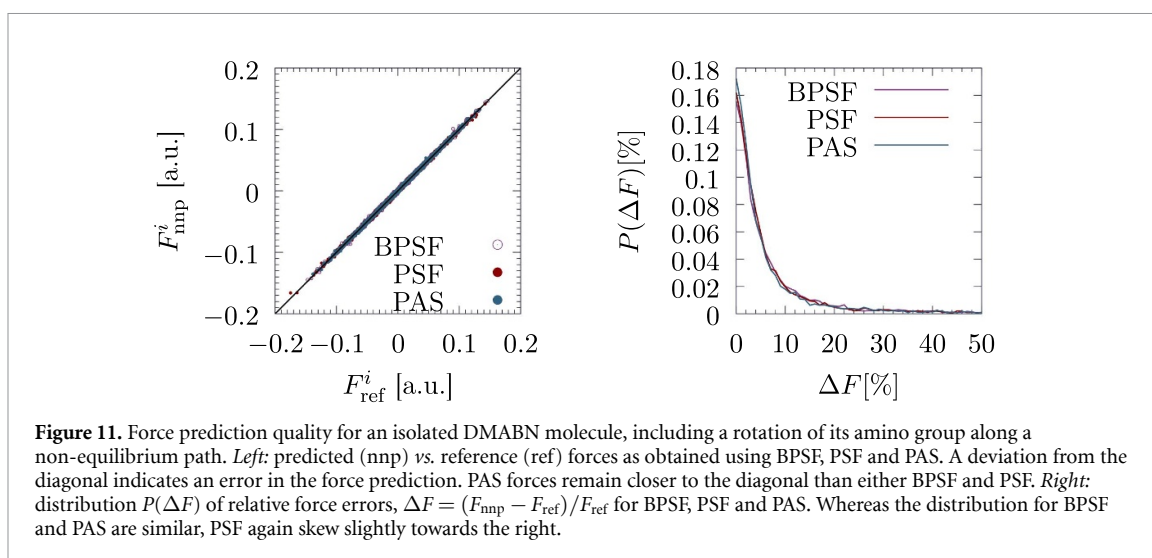


Figure 11. Force prediction quality for an isolated DMABN molecule, including a rotation of its amino group along a non-equilibrium path. *Left:* predicted (nnp) vs. reference (ref) forces as obtained using BPSF, PSF and PAS. A deviation from the diagonal indicates an error in the force prediction. PAS forces remain closer to the diagonal than either BPSF and PSF. *Right:* distribution $P(\Delta F)$ of relative force errors, $\Delta F = (F_{\text{nnp}} - F_{\text{ref}})/F_{\text{ref}}$ for BPSF, PSF and PAS. Whereas the distribution for BPSF and PAS are similar, PSF again skew slightly towards the right.

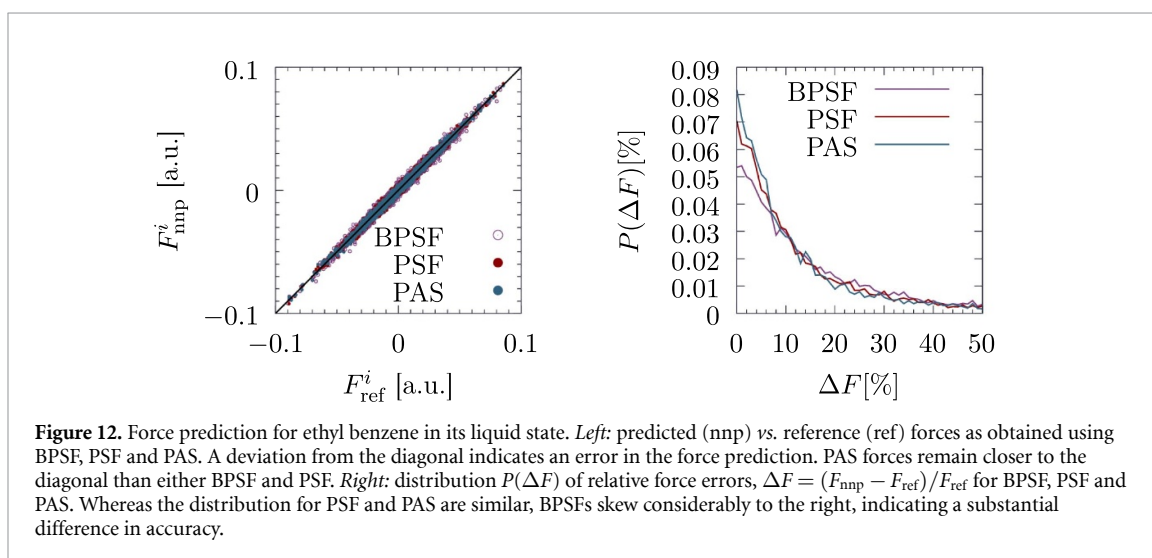


Figure 12. Force prediction for ethyl benzene in its liquid state. *Left:* predicted (nnp) vs. reference (ref) forces as obtained using BPSF, PSF and PAS. A deviation from the diagonal indicates an error in the force prediction. PAS forces remain closer to the diagonal than BPSF and PSF. *Right:* distribution $P(\Delta F)$ of relative force errors, $\Delta F = (F_{\text{nnp}} - F_{\text{ref}})/F_{\text{ref}}$ for BPSF, PSF and PAS. Whereas the distribution for PSF and PAS are similar, BPSFs skew considerably to the right, indicating a substantial difference in accuracy.

predictions obtained here imply that for a generic test set containing a significant amount of off-equilibrium structures, generic PAS and PSF sets are able to outperform BPSFs.

We will now show that the same considerations also hold for an organic liquid in which van der Waals interactions dominate. For a test set of liquid ethyl benzene, using the same SF setups as for DMABN, we find force prediction RMSDs of $89.5 \text{ meV } \text{\AA}^{-1}$ for BPSFs, $68.9 \text{ meV } \text{\AA}^{-1}$ for PSF and $61.7 \text{ meV } \text{\AA}^{-1}$ for PAS, respectively. This corresponds to a difference in accuracy approaching 50% between BPSF and PAS. Whereas these considerable differences are not evident from the diagonal force-to-force plot in the left panel of figure 12, they become visible in the force histogram in the right panel: Histograms for PSF and PAS are very similar, with the only significant difference being the peak around 0%, but BPSFs skew considerably to the right, and much more so than what was observed for PSF in the case of water or Cu_2S . Energy RMSDs are about 0.05 meV/atom for all SFs studied here, which is of excellent accuracy. For liquid anisole, all SF setups result in about equally accurate predictions, as evidenced by figure 13: observed force prediction RMSDs amount to $100.8 \text{ meV } \text{\AA}^{-1}$ for BPSFs and a comparable $106.5 \text{ meV } \text{\AA}^{-1}$ for PSF, with the lowest deviation again being observed at $98.7 \text{ meV } \text{\AA}^{-1}$ for PAS. All SFs yield energy RMSDs that can be considered identical, within a small range from 0.08 meV/atom for BPSF to 0.11 meV/atom for PAS and PSF. Overall, the trends obtained for force predictions in organic molecules indicate that our generic PAS and PSF setups on average exhibit superior transferability between systems, while at the same time retaining lower force errors on the test sets compared to generic BPSFs. Importantly, whereas BPSFs performed considerably worse than polynomial functions for liquid ethyl benzene, neither PAS nor PSF showed a considerably inferior performance with respect to BPSF in any of the other systems investigated here. Energy predictions are much more robust with respect to the choice of system, they remain of comparable accuracy for all SFs investigated here.

Table 1. MAE (kcal mol⁻¹) over training, test and validation set of the QM9 database as described in the main text. Note that the dataset was not normalised.

MAE	Training	Test	Validation
BPSF	1.51	11.21	10.72
PAS	1.63	2.13	2.24

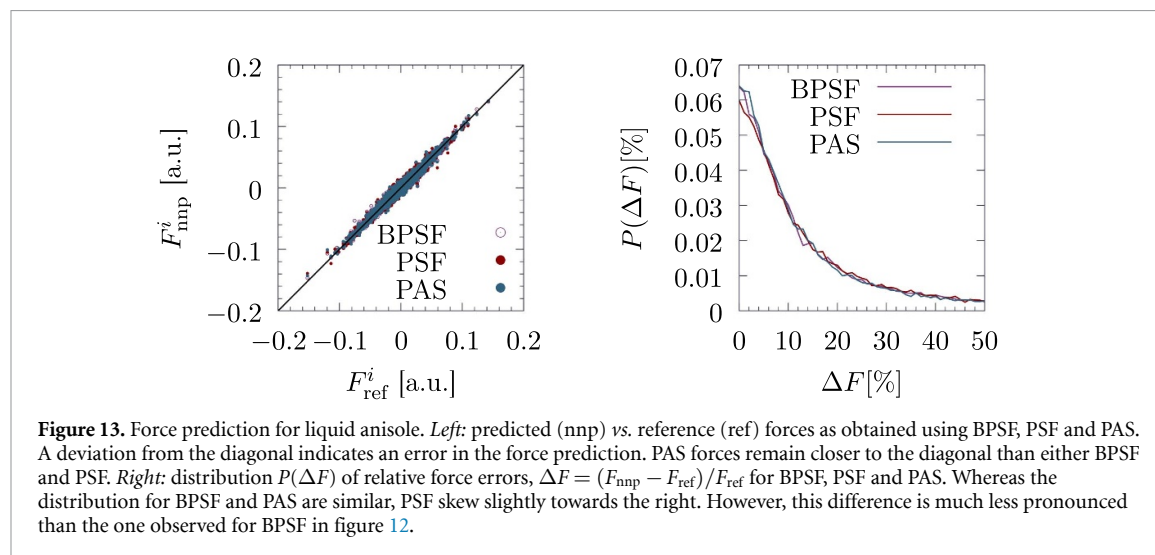


Figure 13. Force prediction for liquid anisole. *Left:* predicted (nnp) vs. reference (ref) forces as obtained using BPSF, PSF and PAS. A deviation from the diagonal indicates an error in the force prediction. PAS forces remain closer to the diagonal than either BPSF and PSF. *Right:* distribution $P(\Delta F)$ of relative force errors, $\Delta F = (F_{\text{nnp}} - F_{\text{ref}})/F_{\text{ref}}$ for BPSF, PSF and PAS. Whereas the distribution for BPSF and PAS are similar, PSF skew slightly towards the right. However, this difference is much less pronounced than the one observed for BPSF in figure 12.

4.4. QM9 database

In order to pinpoint possible improvements when solely energies are to be treated, we have constructed a set of wACSF based on PAS and have used this set to predict enthalpies of formation in the QM9 database. In order to render the training more challenging, and contrary to [21], enthalpies in the dataset were neither normalised nor rescaled. PAS were set up according to the procedure outlined in section 4.3, employing a maximum width of 8 Å and using a random choice of 10 000 structures for training (and testing) as outlined in [21], the remainder being used as a validation set. Mean absolute errors (MAE) over predicted enthalpies of formation for the remaining $\approx 124\,000$ molecules of the database were averaged over 5 independent partitionings into train, test and validation sets as in [21].

As shown in table 1, while performance on the training set is comparable between the best BPSF set reported in [21] and PAS, the quality of the predictions on test and validation sets within a given selection of 10 000 test- and training structures considerably improves when PAS are used. For the random partitionings into training and test data studied here, errors obtained from HDNNPs trained with PAS are about 5-fold smaller than their BPSF counterparts. It should be noted that once BPSF are used to train a HDNNP on an appropriately normalised and rescaled training set, their performance approaches that of PAS on a non-normalised test set [21]. This indicates that the robustness of the predictions substantially increases when PAS are used, as test and validation error are substantially lower even when the dataset is not subjected to any normalisation.

4.5. Computational footprint

The choice of SFs has an influence on both, the computational footprint of training as well as the cost of force predictions during production runs. For the former, if SFs can be calculated for all structures in the training set and then stored, it is mainly the number of SFs that influences performance: more SFs imply more weights to be optimised and as such result in higher execution times. Memory requirements will grow as well as the number of SFs increases. Even if a (generic) set of two SFs might comprise the same number of input functions, pruning of functions that never exceed a certain threshold (typically 10^{-3}) can result in different set sizes during training. In productive MD runs, SFs have to be recalculated for every new structure and hence, performance will not only be determined by their sheer number, but also by the number and type of floating point operations involved in their computation. In the following, we will analyse both performance-related aspects.

Table 2 lists the cost of running 100 steps of MD for a periodic box containing 360 water molecules using the HDNNPs investigated here both for a straightforward calculation of all SFs as well as using an optimised algorithm tailored for distributed memory parallelisation on core processing units (CPUs) as reported in

Table 2. MD simulation execution time of 360 water molecules (100 timesteps, 4 cores). SFGroups/Caching denotes the algorithmic optimisations outlined in [27], values in parentheses denote speedups.

Optimisation	BPSF	PSF	PAS
None	37.28	7.72 (\times 4.83)	8.07 (\times 4.62)
SFGroups/Caching	12.50	6.82 (\times 1.83)	6.90 (\times 1.81)
Speedup via Opt.	\times 2.98	\times 1.13	\times 1.17

Table 3. Number of symmetry functions effectively used during training of the systems investigated here after removing SFs with amplitudes $< 10^{-3}$. BPSF-parameters were chosen according to [24].

System	BPSF	PSF	PAS
Water	57	62	54
Cu ₂ S	138	162	178
DMABN	893	852	841
Ethyl benzene	272	366	368
Anisole	1025	1057	1036

[27]. The former serves as a measure for the cost of the floating-point operations associated to every SF type; this corresponds to the limit of all N out of N SFs being independent and differently parametrised. In comparison with the former, the latter indicates the benefit of underlying algorithmic optimisations; this however requires that a subset of the N SFs contain identical parameters and cutoff functions in order to benefit from a grouping and caching strategy. The effective speedups obtained by using PSF or PAS for the water setup reach almost a factor of 5 when no grouping or caching strategy is applied. This indicates that the number of expensive floating point operations is significantly reduced when purely polynomial SFs are used. Vice versa, SF grouping and caching has a much lower influence of execution times of PSF and PAS, where those optimisations reduce the computational cost by about 15%. As would be expected by the larger number of terms involved in the evaluation of PAS, their computation is slightly more expensive than the evaluation of PSF. However, the difference is small enough to be negligible in practice.

In contrast, BPSFs—for which grouping and caching strategies were developed in the first place—benefit from speedups of up to a factor of 3 when grouping and caching strategies are introduced. Note that, by construction, PAS and PSF are not intended to require use of optimisation strategies, which is reflected in comparably modest speedups upon introduction of grouping and caching. Still, even in those cases, PSF and PAS are almost a factor of 2 faster than BPSF. In addition, it should be noted that PAS yield the lowest RMSD on water with the smallest total number of SFs; they therefore do not only lower the computational cost, but also the memory footprint with respect to BPSFs.

Table 3 compares the numbers of SFs effectively used during training after pruning all SFs that never exceed a magnitude of 10^{-3} . This provides a measure for computational cost and memory requirements. Apart from ethyl benzene, in which significantly more SFs of BPSF-type were removed during pruning, numbers are comparable between BPSF, PSF and PAS. It should be noted that all systems except water were trained with automatically generated sets of SF parameters, which also accounts for the different extent of functions with maximum amplitude of $< 10^{-3}$ in these setups. In this context, it is interesting to note that in spite of the structural similarities between anisole and ethyl benzene, for BPSFs, there is a significant difference in the number of pruned SFs. For PAS and PSF, no such difference can be observed; this can also account for the larger errors in force RMSDs that are observed when ethyl benzene is trained with a set of BPSFs. Within the hand-picked set of SFs for water, differences between PSF and PAS are notable, with the latter allowing for a reduction of about 15% with respect to the former. BPSFs lie in between. This suggests that, for hand-selected sets, PAS allow for an even smaller number of SFs to be used.

Overall, the performance of PSF and BPSFs was comparable for all systems studied here. Force RMSDs for water and Cu₂S were slightly higher for PSF; however, this was not reflected in the dynamic and structural properties of water obtained by 2 ns of HDNNP-MD, where results using PSF and BPSFs showed excellent agreement. On the other hand, for DMABN and liquid ethyl benzene, PSF considerably outperformed BPSFs. For all systems, PAS were able to outperform either PSF or BPSF. In particular, when used in the wACSF scheme [21], PAS greatly improve performance over BPSF for data sets that are not normalised, indicating superior robustness. We therefore strongly recommend PAS for future HDNNP setups, since they allow to speed up productive MD runs by a factor of 1.8 and further improve accuracy of the HDNNPs. In particular, due to their floating point cost being lower by about a factor of almost 5, we particularly

recommend use of PAS for implementations where caching and grouping strategies are not feasible. This can, for instance, be the case when offloading computation of SFs to graphical processing units, where the resolution of if-statements and loop breaks associated to optimisation schemes can considerably slow down computation with respect to purely arithmetic operations. Not least, thanks to their small computational footprint, PAS can also be used in programs that have not undergone extensive optimisation, and their simple structure facilitates implementation and calculation of derivatives up to an arbitrary order defined by the underlying polynomial.

5. Conclusion and outlook

Here, we have introduced a new family of SFs for Behler–Parrinello HDNNPs, based on polynomials with compact support for both radial and angular environments, which supersedes use of a separate set of radial cutoff functions. The centres and widths of our PSFs can be freely chosen. This is notably the case for angular functions, which have so far been restricted to peak at 0° or 180° . As long as symmetry of derivatives on 0° and 180° is maintained, angular functions can be centred on any point within $[0^\circ, 180^\circ]$ and their width $\delta\theta$ can be freely chosen. We have introduced two types of radial SFs, PSF and PAS, with the former being point-symmetric on $[r_s, r_s + \Delta]$, whereas the latter have a long-range tail reminiscent of the product of a cutoff function and a Gaussian. Our PSFs considerably simplify the choice of radial SF parameters with respect to common Gaussian functions, since the position of maxima and minima can be straightforwardly predicted without having to take into account shifts and asymmetries introduced by a cutoff function.

The accuracy of a PSF- and PAS-based setup with respect to conventional BPSFs was assessed for various phases of water, for solid Cu_2S as well as for organic molecules in the gaseous and liquid phase. For water, Cu_2 and DMABN, PSF, PAS and BPSF all showed excellent accuracy. Structural and dynamic properties obtained from 2 ns HDNNP-MD of liquid water revealed no relevant differences between the NNPs obtained with any of the three SF types. In the case of off-equilibrium structures of a DMABN molecule in the gas phase, PAS and PSF have outperformed BPSFs for force predictions by 15% and improved energy RMSDs by 50%. This situation was much more prominent for liquid ethyl benzene, where force RMSDs over the test set were more than 50% higher for BPSF than for PSF or PAS. Force prediction errors were generally higher for liquid anisole, however, they remained highly comparable between all types of SFs studied here. Generally, PSF performed on par with BPSF, except for ethyl benzene where they performed considerably better. For all systems studied here, PAS consistently yielded the best results. By using PAS in wACSFs, the predictive power of a HDNNP trained on 10 000 non-normalised structures of the QM9 database was considerably improved, with mean absolute errors for training, test and validation set being about five-fold lower than for BPSF-based wACSFs.

We have demonstrated that HDNNP-MD of liquid water is about two times faster using PSF or PAS compared to BPSE, constituting a considerable improvement of performance and hence, sampling. In terms of the floating point operations associated to the evaluation of SFs, we found that use of PSF results in speedups of a factor of about 5 with respect to BPSE, and speedups associated to the slightly more complex PAS still exceed a factor of 4.5. The number of SFs used for the systems investigated here were comparable between PSF, PAS and BPSE, although results for the hand-picked set for water indicate that setups with PAS can potentially reduce the number of SFs used with respect to common BPSFs, therefore saving on memory and on the cost of training by reducing the number of fitting parameters. Given the substantial reduction in extensive floating point operations associated to the use of PSF and PAS, and given that accuracy of generic sets of PAS is consistently better than that of BPSE, we advocate use of PAS as a method of choice for future implementations. In particular, their simple structure not only simplifies their implementation and makes their setup straightforward, but their small computational footprint can also facilitate their porting to different system architectures without the need for extensive optimisation.

Overall, we have shown that by constructing SFs with compact support, substantial performance gains of up to a factor of almost 5 can be obtained without impacting accuracy of the resulting HDNNPs. Polynomial SFs are easy to implement and set up, since they allow for a flexible choice of radial and angular domain do not require any cutoff functions. Our PSFs therefore lend themselves for all applications of HDNNPs, be it for equilibrium or off-equilibrium structures of isolated molecules, liquids or solids.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI: <https://github.com/CompPhysVienna/n2p2> [31].

Acknowledgments

M P B acknowledges a postdoctoral fellowship from the Swiss National Science Foundation (Project No. 184500). A S acknowledges support from the European Union's Horizon 2020 research and innovation programme under Grant Agreement No. 676531 (project E-CAM). The results presented here were in part obtained using the Vienna Scientific Cluster (VSC).

ORCID iDs

Martin P Bircher  <https://orcid.org/0000-0002-6905-3130>
Andreas Singraber  <https://orcid.org/0000-0002-4330-1394>
Christoph Dellago  <https://orcid.org/0000-0001-9166-6235>

References

- [1] Dral P O 2020 Quantum chemistry in the age of machine learning *J. Phys. Chem. Lett.* **11** 2336–47
- [2] Mater A C and Coote M L 2019 Deep learning in chemistry *J. Chem. Inf. Model.* **59** 2545–59
- [3] Butler K T, Davies D W, Cartwright H, Isayev O and Walsh A 2018 Machine learning for molecular and materials science *Nature* **559** 547–55
- [4] Behler J 2016 Perspective: machine learning potentials for atomistic simulations *J. Chem. Phys.* **145** 170901
- [5] Morawietz T, Singraber A, Dellago C and Behler J 2016 How van der Waals interactions determine the unique properties of water *Proc. Natl Acad. Sci. USA* **113** 8368–73
- [6] Cheng B, Engel E A, Behler J, Dellago C and Ceriotti M 2019 Ab initio thermodynamics of liquid and solid water *Proc. Natl Acad. Sci. USA* **116** 1110–5
- [7] Senftle T P et al 2016 The ReaxFF reactive force-field: development, applications and future directions *npj Comput. Mater.* **2** 1–14
- [8] Barcaro G, Carravetta V, Sementa L and Monti S 2019 Reactive force field simulations of silicon clusters *Adv. Phys. X* **4** 1634487
- [9] Deringer V L, Caro M A and Csányi G 2019 Machine learning interatomic potentials as emerging tools for materials science *Adv. Mater.* **0** 1902765
- [10] Faber F A et al 2017 Prediction errors of molecular machine learning models lower than hybrid DFT error *J. Chem. Theory Comput.* **13** 5255–64
- [11] Bartók A P, Kondor R and Csányi G 2013 On representing chemical environments *Phys. Rev. B* **87** 184115
- [12] Bartók A P, Payne M C, Kondor R and Csányi G 2010 Gaussian approximation potentials: the accuracy of quantum mechanics, without the electrons *Phys. Rev. Lett.* **104** 136403
- [13] Rupp M, Tkatchenko A, Müller K-R and von Lilienfeld O A 2012 Fast and accurate modeling of molecular atomization energies with machine learning *Phys. Rev. Lett.* **108** 058301
- [14] Thompson A P, Swiler L P, Trott C R, Foiles S M and Tucker G J 2015 Spectral neighbor analysis method for automated generation of quantum-accurate interatomic potentials *J. Comput. Phys.* **285** 316–30
- [15] Blank T B, Brown S D, Calhoun A W and Doren D J 1995 Neural network models of potential energy surfaces *J. Chem. Phys.* **103** 4129–37
- [16] Behler J 2011a Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations *Phys. Chem. Chem. Phys.* **13** 17930–55
- [17] Behler J and Parrinello M 2007 Generalized neural-network representation of high-dimensional potential-energy surfaces *Phys. Rev. Lett.* **98** 146401
- [18] Behler J 2015 Constructing high-dimensional neural network potentials: a tutorial review *Int. J. Quantum Chem.* **115** 1032–50
- [19] Behler J 2011b Atom-centered symmetry functions for constructing high-dimensional neural network potentials *J. Chem. Phys.* **134** 074106
- [20] Geiger P and Dellago C 2013 Neural networks for local structure detection in polymorphic systems *J. Chem. Phys.* **139** 164105
- [21] Gastegger M, Schwiedrzik L, Bittermann M, Berzsenyi F and Marquetand P 2018 wACSF—weighted atom-centered symmetry functions as descriptors in machine learning potentials *J. Chem. Phys.* **148** 241709
- [22] Morawietz T and Behler J 2013a A density-functional theory-based neural network potential for water clusters including van der Waals corrections *J. Phys. Chem. A* **117** 7356–66
- [23] Morawietz T and Behler J 2013b A full-dimensional neural network potential-energy surface for water clusters up to the hexamer *Z. Phys. Chem.* **227** 1559–81
- [24] Imbalzano G, Anelli A, Giofré D, Klees S, Behler J and Ceriotti M 2018 Automatic selection of atomic fingerprints and reference configurations for machine-learning potentials *J. Chem. Phys.* **148** 241730
- [25] Smith J S, Isayev O and Roitberg A E 2017 ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost *Chem. Sci.* **8** 3192–203
- [26] Singraber A, Morawietz T, Behler J and Dellago C 2019 Parallel multistream training of high-dimensional neural network potentials *J. Chem. Theory Comput.* **15** 3075–92
- [27] Singraber A, Behler J and Dellago C 2019 Library-based LAMMPS implementation of high-dimensional neural network potentials *J. Chem. Theory Comput.* **15** 1827–40
- [28] Bircher M P and Rothlisberger U 2018 Plane-wave implementation and performance of à-la-Carte Coulomb-attenuated exchange-correlation functionals for predicting optical excitation energies in some notorious cases *J. Chem. Theory Comput.* **14** 3184–95
- [29] Peach M J G, Benfield P, Helgaker T and Tozer D J 2008 Excitation energies in density functional theory: an evaluation and a diagnostic test *J. Chem. Phys.* **128** 044118
- [30] Ramakrishnan R, Dral P O, Rupp M and von Lilienfeld O A 2014 Quantum chemistry structures and properties of 134 kilo molecules *Sci. Data* **1** 140022

- [31] Singraber A *et al* 2018 *N2p2: NeuralNetworkPotentialPackage* CompPhysVienna (Available at: <https://github.com/CompPhysVienna/n2p2/>) (<https://doi.org/https://doi.org/10.5281/zenodo.1344446>)
- [32] CPMD consortium 2000 CPMD Jointly by IBM Corp. and by Max Planck Institute Stuttgart
- [33] Sun J, Ruzsinszky A and Perdew J P 2015 Strongly constrained and appropriately normed semilocal density functional *Phys. Rev. Lett.* **115** 036402
- [34] Plimpton S 1995 Fast parallel algorithms for short-range molecular dynamics *J. Comput. Phys.* **117** 1–19
- [35] Plimpton S, Thompson A, Moore S and Kohlmeyer A 2004 LAMMPS Molecular Dynamics Simulator