

PAPER • OPEN ACCESS

Measuring transferability issues in machine-learning force fields: the example of gold–iron interactions with linearized potentials

To cite this article: Magali Benoit *et al* 2021 *Mach. Learn.: Sci. Technol.* **2** 025003

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



PAPER

OPEN ACCESS

RECEIVED

2 October 2020

REVISED

28 October 2020

ACCEPTED FOR PUBLICATION

12 November 2020

PUBLISHED

28 December 2020

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.



Measuring transferability issues in machine-learning force fields: the example of gold–iron interactions with linearized potentials

Magali Benoit¹, Jonathan Amodeo², Ségolène Combettes¹, Ibrahim Khaled³, Aurélien Roux³ and Julien Lam³

¹ CEMES, CNRS and Université de Toulouse, 29 rue Jeanne Marvig, 31055 Toulouse Cedex, France

² Université de Lyon, INSA-Lyon, MATEIS, UMR 5510 CNRS, 69621 Villeurbanne, France

³ Center for Nonlinear Phenomena and Complex Systems, Université Libre de Bruxelles, Code Postal 231, Boulevard du Triomphe, 1050 Brussels, Belgium

E-mail: julien.lam@ulb.ac.be

Keywords: machine-learning interaction potential, LassoLars, gold–iron, transferability, nanoparticle

Abstract

Machine-learning force fields have been increasingly employed in order to extend the possibility of current first-principles calculations. However, the transferability of the obtained potential cannot always be guaranteed in situations that are outside the original database. To study such limitation, we examined the very difficult case of the interactions in gold–iron nanoparticles. For the machine-learning potential, we employed a linearized formulation that is parameterized using a penalizing regression scheme which allows us to control the complexity of the obtained potential. We showed that while having a more complex potential allows for a better agreement with the training database, it can also lead to overfitting issues and a lower accuracy in untrained systems.

1. Introduction

Atomistic modeling is often divided in two different types of simulations. On the one hand, quantum methods including Hartree–Fock and density functional theory (DFT) approaches are considered the most accurate and are employed for virtually any types of chemical species [1, 2]. On the other hand, classical force fields are used to perform large-scale and long-time simulations with less accuracy [3, 4]. However, it is still difficult to connect both approaches and until now, one can hardly perform a simulation involving millions of atoms for nanoseconds while retaining the accuracy of quantum methods.

In this context, machine-learning interaction potentials (MLIP) have been proposed in the recent years and have shown great potentials to achieve such simulations [5–7]. Numerous approaches are currently considered including artificial neural networks [8], Gaussian approximation methods [9], linearized potentials [10, 11], spectral neighbor analysis potential [12], symmetric gradient domain machine learning [13, 14] and moment tensor potentials [15]. The success of these techniques is recognized by the large variety of materials that have been successfully tackled: pure metals [16–20], organic molecules [21–24], oxides [25, 26], water [27–31], amorphous materials [32–37] and hybrid perovskites [38].

For all of these techniques, the main procedure consists in using a very universal analytical formulation for the force field which is then parameterized to match a database of DFT calculations including total energy, forces and stress tensors. However, it is admitted that MLIP can sometimes show poor transferability towards systems that are not included in the learning database. In the worst scenario, the MLIP is so-well fitted to its learning database that non-physical behaviors may be observed outside of it. In order to fix this issue, the main proposal is to regularly check the accuracy of the potential as the machine-learning molecular dynamics simulations are carried out and to improve the MLIP ‘on the fly’ [38–40]. Yet, to the best of our knowledge, such flaw of the approach has never been quantitatively investigated while being acknowledged by both users and developers.

For our case study, we choose interactions in gold–iron nanoparticles. In principle, such system can be found concurrently in three different chemical orderings namely ‘alloy’, ‘Janus’ and ‘core-shell’. Yet, recent experiments have shown that the synthesized Au–Fe nanoparticles are made of an iron core wrapped in a gold shell and that the shape of the iron core depends strongly on the amount of surrounding gold [41–45]. These nanoparticles have potential biomedical applications as iron is known for its intrinsic ferromagnetism and gold capping can protect the iron core from oxidation. However, rationalizing the results of the synthesis along with predicting the material properties would require numerical simulations which are sparse for gold–iron nanoparticles [45–49]. Indeed, while full quantum calculations can not be employed to study clusters of more than tens of atoms, the empirical modeling of gold–iron nanoparticles is also a very difficult case because these two metals are non miscible at room temperature on a very large domain of the phase diagram. There are therefore no iron–gold alloy crystal structures to adjust the parameters of an empirical potential. Moreover, iron is magnetic and crystallizes in a bcc structure while gold crystallizes in an fcc structure, which makes the development of a potential capable of capturing all the properties of this alloy even more complex. Previous attempts have shown their limits by stabilizing metastable alloys [50] or by not being able to find the most stable Fe/Au interfaces [46, 47], leading us to develop potentials specifically dedicated to a particular problem, and hence highly non-transferable [49].

In this article, we begin by describing the methodology including linearized machine-learning potential and a penalizing regression scheme. In the results section, we first studied the influence of the descriptor functions. Then, we showed that the methodology allows one to quickly obtain MLIPs with different degrees of complexity. Afterwards, the transferability of these different potentials was tested on forces in untrained chemical orderings namely Janus and core–shell. While the error should decrease monotonically when increasing the MLIP complexity, we observed a surprising non-monotonic behavior thus illustrating that more complexity does not necessarily lead to a better MLIP overall. Such transferability issue was reduced by using a more diverse set of descriptors. Finally, we measured some properties of the bulk and investigated the possibilities and the limitations of the obtained MLIP toward bulk simulations even if it was trained on nanostructures.

2. Methods

2.1. The Φ -Lassolars machine-learning interaction potential

For our MLIP, we employed the analytical formulation originally put forward by Seko *et al* [10, 17, 18, 51, 52]. In this method, the total potential energy of a configuration made of N atomic positions is first given by $E_{\text{tot}} = \sum_{i=0}^N E_i$ where E_i is the atomic energy. For E_i , we considered a weighted linear combination of descriptors indexed by n :

$$E^{(i)} = \sum_n \omega_n X_n^{(i)}, \quad (1)$$

where ω_n is the linear coefficient associated with the descriptor $X_n^{(i)}$. Until now, moment tensors [16], group-theoretical high-order rotational invariants [52] and bispectrum components [11, 12, 53] were previously proposed as descriptors for such linearized potentials. In this work, we favored a simpler formulation which consists in developing the descriptor space in explicit two-body, three-body and N-body interactions:

$$[2B]_n^{(i)} = \sum_j f_n(R_{ij}) \times f_c(R_{ij}), \quad (2)$$

$$[3B]_{(n,l)}^{(i)} = \sum_j \sum_k f_n(R_{ij}) f_c(R_{ij}) f_n(R_{ik}) f_c(R_{ik}) \cos^l(\theta_{ijk}), \quad (3)$$

$$[\text{NB}]_{(n,m)}^{(i)} = \left[\sum_j f_n(R_{ij}) \times f_c(R_{ij}) \times f_s(R_{ij}) \right]^m, \quad (4)$$

where R_{ij} is the distance between atoms i and j , θ_{ijk} is the angle centered around the atom i , and l and m are two positive integers. For the cut-off function, we chose what was originally proposed by Behler and

Parrinello [8]: $f_c = \frac{1}{2} (1 + \cos(\pi(R_{ij}/R_{\text{cut}})))$ with R_{cut} being set at 6 Å. The switch function denoted f_s is employed in order to prevent from non-physical behavior of the N-body contribution at short distances. To do so, two distances r_1 and r_2 are first defined as respectively 95% and 105% of the minimum of the dimer interactions and then a function is constructed to smoothly go from 0 to 1 in the range of $[r_1:r_2]$: $f_s(u) = 6u^5 - 15u^4 + 11^3$ where $u = (R_{ij} - r_1)/(r_2 - r_1)$ [54]. Altogether, these expressions allow for a direct computation of the forces as well as the stress tensors by differentiating with respect to the positions [10]. Regarding the basis of functions f_n , there are no physical restrictions. In particular, for the two-body interactions, one can tune f_n to mimic traditional interatomic potentials as for example Morse, Lennard–Jones, Buckingham or Yukawa potentials or use simple functions like Gaussians, Lorentzian or asymmetric log-normal functions. Likewise, for the three-body interactions, the current formulation is very similar to what is done in the Stillinger–Weber potential [55]. Finally, the N-body interaction is a generalized form of the EAM potential where the embedding function is a polynomial of the atomic density [56]. In this case, the integer m corresponds directly to the degree of N-body order. The difference between our formulation and the most recent ones proposed by Seko *et al* is that, in the N-body interactions, we did not include any explicit angular dependence and did not mix different forms of the atomic density.

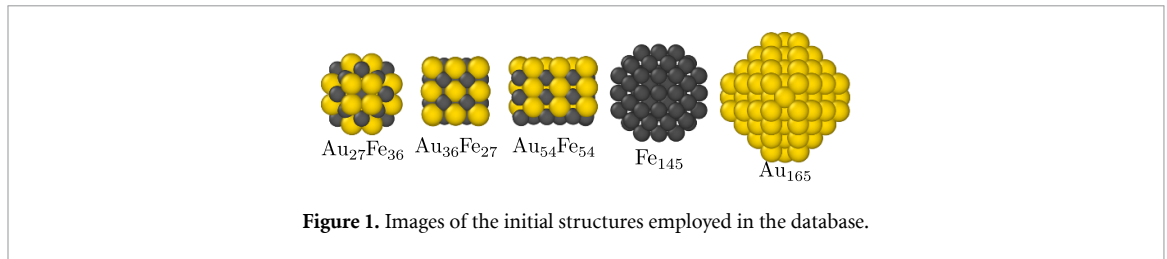
For the fitting procedure of such a linear model, previous studies have proposed the use of genetic algorithm [12, 53], weighted ordinary least squares [11], Bayesian linear regression [39], ridge regression [18, 51, 52] and least absolute shrinkage and selection operator (Lasso) [17, 10]. In order to construct a simpler MLIP, we employed the Lasso regression with the Least Angle Regression Scheme (Lars, together denoted LassoLars) [57]. In practice, along with the ordinary least square objective function, χ_{OLS}^2 , the Lasso scheme adds a penalty on the sum over the absolute value of the coefficients ω_n and the employed error function is therefore given by

$$\chi^2 = \chi_{\text{OLS}}^2 + \alpha \sum_n |\omega_{(n)}|, \quad (5)$$

where α is a parameter that controls the degree of penalty. The penalty on the absolute value of the coefficients enforces lots of the linear coefficients to be exactly 0. Additionally, using Lars allows us to select the most relevant descriptors by measuring their correlation to the target. Using LassoLars as a regression scheme is at the expense of accuracy and flexibility for the MLIP but it allows for a considerable reduction in the complexity of the potential. Please see the appendix for further motivation regarding the choice of LassoLars and for additional details on the Φ -LassoLars implementation.

2.2. DFT database

Building a general purpose potential for Au–Fe nano-crystals would require to model the atomic interactions not only in different structures (crystal polymorph, interfaces and liquid for instance) but also in different chemical orderings including alloyed, Janus and core–shell nanoparticles. Because the focus of this work is to measure the issues related to transferability, we purposely employed an incomplete database made of only three types of nanoparticles: (1) alloys with almost equimolar compositions, (2) pure iron in the body-centered cubic (bcc) phase, (3) pure gold in the face centered cubic (fcc) phase (see figure 1). Then, molecular dynamics (MD) simulations by means of a house code were carried out to melt the constructed nanoparticles. Simulations were performed in the NVT ensemble obtained with Andersen thermostat at 1400 K during 500 ps using a timestep of 1 fs. We used simple pair-wise potentials made of Lennard–Jones and Morse interactions for gold and iron respectively and of Lennard–Jones interaction for the gold–iron cross-interaction. The employed Morse parameterization is the one of Hung *et al* [58], while the two Lennard–Jones potentials for gold–gold and iron–gold were simply parameterized in order to match bulk lattice parameters and cohesive energies. Along the melting path, we extracted configurations that are representative of the solid to liquid transition. In addition, each initial structure was also manually compressed. The distances are reduced by a factor of 75% and along the compression, we extracted 10 configurations in order to sample structures of higher density and better reproduce the repulsion at short distances. For the same reason, diatomic molecules FeFe, AuFe and AuAu with distances down to 1 Å were also added in the database. For each of these configurations, forces were finally computed at the DFT-level using single-point calculations. Spin-polarized DFT calculations were performed with the VASP code [59], using PAW type pseudopotentials for iron and gold [60], a plane wave cutoff of 650 eV and a Methfessel-Paxton smearing parameter σ of 0.01 eV. All calculations were done at the Γ -point of the Brillouin zone. Altogether, the database is made of 181653 atomic configurations with an almost equal proportion in the three types of nanoparticles (34% of alloy, 34% of pure gold and 32% of pure iron). We note that the database sampling was made with classical interaction potentials which is less satisfying than performing *ab initio* MD. Yet, the advantage is that it allows us to quickly sample configurations that remain

**Table 1.** Summary of the tested descriptors.

Function name	Equation	List of a_n	List of b_n	N_{func}
Gaussian	$f_n(R_{ij}) = \exp(-a_n(R_{ij} - b_n)^2)$	0.5, 1.0, 1.5	1, 2, 3, 4, 5, 6	990
Lorentzian peak	$f_n(R_{ij}) = 1/((R_{ij} - a_n)^{b_n} + 1)$	1, 2, 3, 4, 5, 6	2, 4, 6, 8	1320
Log-normal peak	$f_n(R_{ij}) = \exp(-\ln[(R_{ij} - a_n)/b_n]^2)$	1, 2, 3, 4, 5, 6	1.0, 1.5, 2.0, 2.5	1320
Slatter-type orbital	$f_n(R_{ij}) = R_{ij}^{a_n} \exp(-b_n R_{ij})$	-2, -1, 0, 1, 2	2, 4, 6, 8, 10	1375
Gaussian-type orbital	$f_n(R_{ij}) = R_{ij}^{a_n} \exp(-b_n R_{ij}^2)$	-2, -1, 0, 1, 2	2, 4, 6, 8, 10	1375

physically valid and it prevents from running very computationally demanding simulations for those relatively large nanoparticles (more than 100 for the alloy and the pure nanoparticles). In addition, by employing configurations that span from the crystalline to the liquid regime, we assure a large variety of atomic neighborhoods with forces ranging from 10^{-4} to $5 \text{ eV } \text{\AA}^{-1}$. Finally, during the manual compression of the nanoparticle and when computing the forces for diatomic molecules, it may happen that some atoms were very close thus leading to very large forces. In the χ^2 , those large forces would contribute a lot while being very unlikely to emerge in a realistic dynamics. Therefore, before performing the fitting procedure, the database is filtered out to remove cases where the forces are larger than $5 \text{ eV } \text{\AA}^{-1}$. Such value was chosen large enough to keep some part of the repulsive interaction but not too large to avoid a fitting focusing only on the repulsive part. Within the fitting procedure, the database is randomly divided in a training (95%) and a test (5%) sets.

3. Results

3.1. Influence of the descriptor space

First, five different types of descriptors were tested. In particular, we used three functions that are ‘peak’ functions i.e. Gaussian, Lorentzian and Log-normal peaks and two functions that are usually employed for orbital calculations and that diverge at short distance i.e. Slatter-type (STO) and Gaussian-type (GTO) orbital (see table 1). Then, the LassoLars method is employed to obtain five different interaction potentials using $\alpha = 10^{-7}$. For the three-body and the N-body interactions, we used respectively $l = [1, 2, 3, 4, 5]$ and $m = [4, 5, 6, 7]$. Altogether, we have around a thousand available descriptors which include Au–Au, Fe–Fe and Fe–Au interactions for each type of descriptors. In figure 2(a), we show the fitting error measured as the root mean square error (RMSE) for the five different types of descriptors. RMSE measured on training and test data sets are similar which means that at this stage, no overfitting is observed. In addition, it appears that the two orbital functions are not as good as the peak functions although STO still gives an RMSE equal to $0.17 \text{ eV } \text{\AA}^{-1}$. Gaussian and Lorentzian descriptors are able to reproduce the forces with an RMSE respectively equal to 0.13 and $0.14 \text{ eV } \text{\AA}^{-1}$. Such values are similar to what is obtained with the generally employed MLIP methods including neural networks, Gaussian approximation method and linearized potentials [61]. In figure 2(b), the number of non-zero coefficient is plotted for the five different types of descriptors. In the cases of GTO and STO functions, much fewer descriptors were selected in comparison to the peak functions. In overall, it remains that the LassoLars algorithm allows one to drastically decrease the number of employed descriptors with respect to the number of available descriptors.

In addition, before studying the transferability issues, we wish to illustrate a second advantage of using the LassoLars algorithm which consists in having a penalizing parameter that controls both the accuracy and the complexity of the obtained potential. According to figure 3, by increasing α , the number of non-zero coefficients and the computational cost can be reduced at the expense of increasing the RMSE. As such, with the LassoLars algorithm, one can simply choose which degree of accuracy or complexity is required for their usage. Finally, the presence of a plateau for the smallest values of α shows that the LassoLars regression only selects relevant descriptors thus reducing the potential complexity. Similar to what was obtained previously using $\alpha = 10^{-7}$, we note that: (1) the 3 peak functions are the most accurate and behave similarly and (2) the

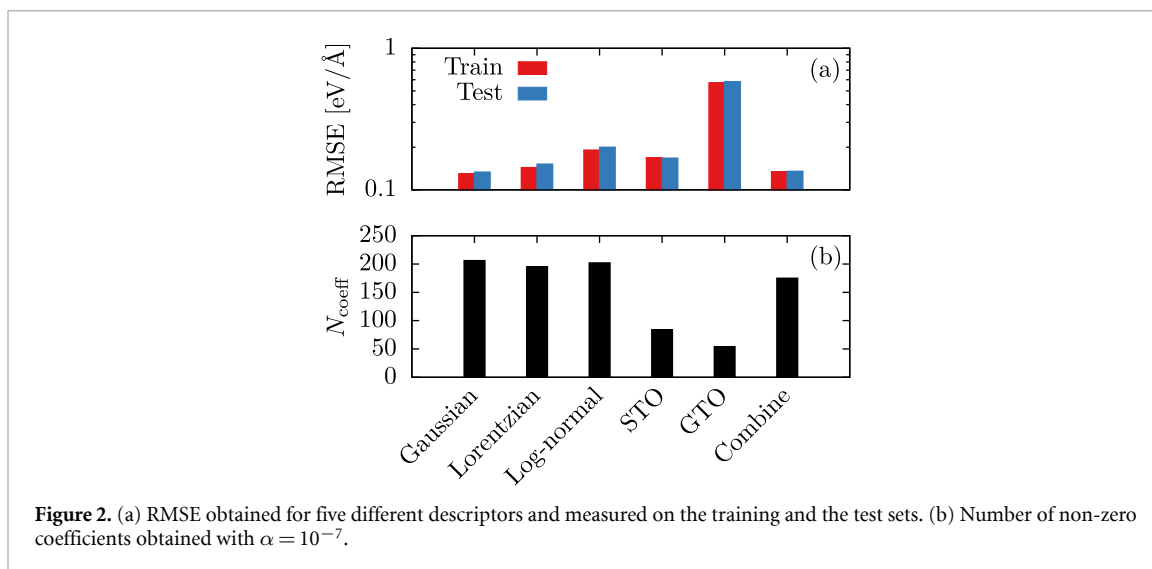


Figure 2. (a) RMSE obtained for five different descriptors and measured on the training and the test sets. (b) Number of non-zero coefficients obtained with $\alpha = 10^{-7}$.

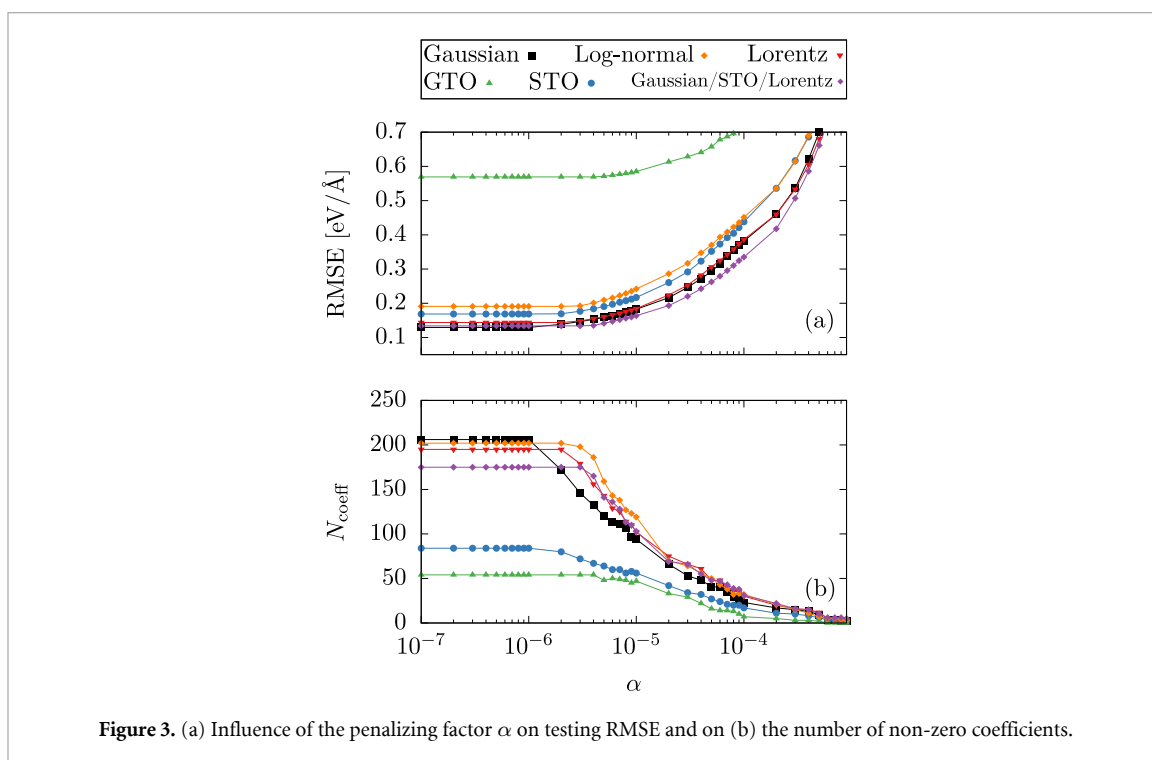


Figure 3. (a) Influence of the penalizing factor α on testing RMSE and on (b) the number of non-zero coefficients.

STO function gives slightly higher RMSE yet with much fewer non-zero coefficients. Figure 3 evidences that the LassoLars algorithm gives the ability to finely control the complexity of the potential at the expense of the accuracy on the DFT database. In the following, we will test how this complexity can influence the MLIP transferability.

3.2. Complexity vs transferability

For that purpose, three additional morphologies of gold-iron nanoparticles were designed: two Janus and one core-shell (see figure 4(a)). Being able to accurately retrieve the interactions in those structures is a difficult test for the MLIP as the training set did not possess any of those demixed structures. Figure 4(b) shows the corresponding RMSE on the forces without having trained the potential on these structures and using only Gaussian functions. Figure 4(c) shows that most of the errors are located at the gold/iron interface which was not included in the training database. Surprisingly, the RMSE behavior is non-monotonic with a minimum located for the three structures around $\alpha = 8 \times 10^{-6}$ which corresponds to 90 non-zero coefficients. More specifically, when transferring the obtained potential to Fe100Au100 Janus nanoparticles, the RMSE obtained with $\alpha = 10^{-7}$ is 30% higher than with the less complex potential that was obtained with $\alpha = 8 \times 10^{-6}$. Therefore, while increasing the complexity of the potential leads to a better agreement within

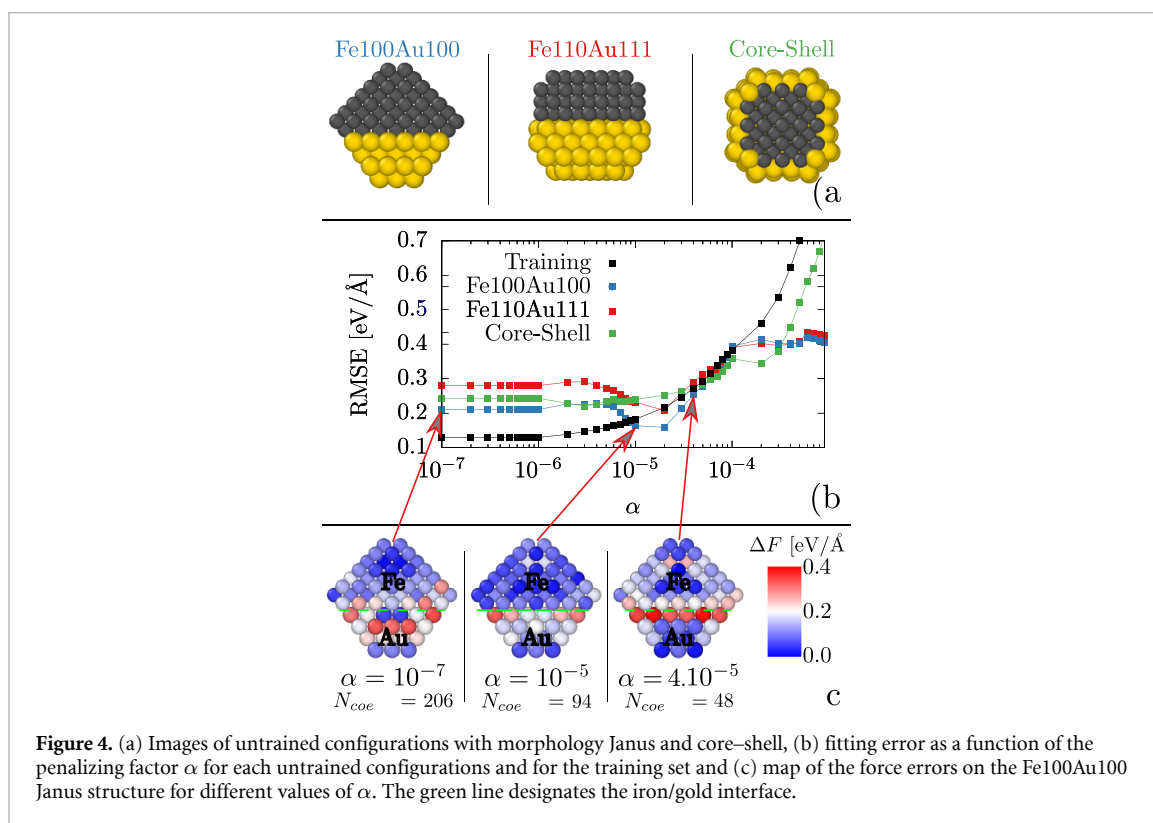


Figure 4. (a) Images of untrained configurations with morphology Janus and core-shell, (b) fitting error as a function of the penalizing factor α for each untrained configurations and for the training set and (c) map of the force errors on the Fe100Au100 Janus structure for different values of α . The green line designates the iron/gold interface.

the training database, it does not necessarily lead to an improvement of the RMSE in unlearned structures using in our case different chemical ordering (i.e. Janus and core-shell instead of alloy nanoparticles). Our challenging test demonstrates that precautions should be made when using machine-learning approaches and that increasing the complexity does not automatically lead to a better overall potential.

According to figure 5, the non-monotonic behavior that was highlighted when using only Gaussian functions is also observed with the two other peak functions i.e. Lorentzian and Log-normal. We also note that, GTO is again very inaccurate and should most probably be avoided for usage as descriptor. Finally, the STO functions while being less accurate on the training database seems to give an overall better results in terms of transferability.

In closing, a combination of three different descriptors (Gaussian, Gaussian-type orbital and Lorentzian peak) is tested in order to employ simultaneously two different types of peak functions along with a function that diverge at short repulsion. STO was chosen for its remarkable ability to decrease the number of non-zero coefficient while Lorentzian peaks also showed slightly less non-monotonicity. On the one hand, regarding its fitting performance, this combination gives an RMSE similar to that obtained previously yet with fewer non-zero coefficients (see figure 5). Having these two additional functions gives more flexibility in the descriptor space and fewer functions can therefore be selected for the same accuracy. On the other hand, for the transferability towards the unlearned structures, the combination gives RMSE that are comparable to the peak functions and even better in two cases (i.e. Fe110Au111 and core-shell). More importantly, it allows for a further reduction of the non-monotonicity. We note that being able to combine three different types of descriptors at the same time is an additional advantage of using a constrained linear regression scheme such as LassoLars.

3.3. Quality of the potential

Even if the aim of the paper was not to obtain an all-purpose MLIP for gold-iron interactions, we still wish to assess the quality of the obtained potential. Regarding the force accuracy on the learned database, figures 6(a) and (b) shows the correlation plot obtained with two different values of α . For the RMSE, we obtained in both cases values of $0.14 \text{ eV } \text{\AA}^{-1}$ when using the three descriptors at the same time. Such value is on par with most of the currently employed MLIP methods [27] and by comparison, the EAM potential that was recently developed for Au-Fe nanoparticle [62] gives a value of $1.4 \text{ eV } \text{\AA}^{-1}$. Our MLIP is thus already a drastic improvement in force evaluation for the studied nanoparticles (see figure 6(c)).

Furthermore, additional simulations with the MLIP were performed to check some properties in the bulk phase although it was not included in the training set. Simulations were carried out using the large-scale

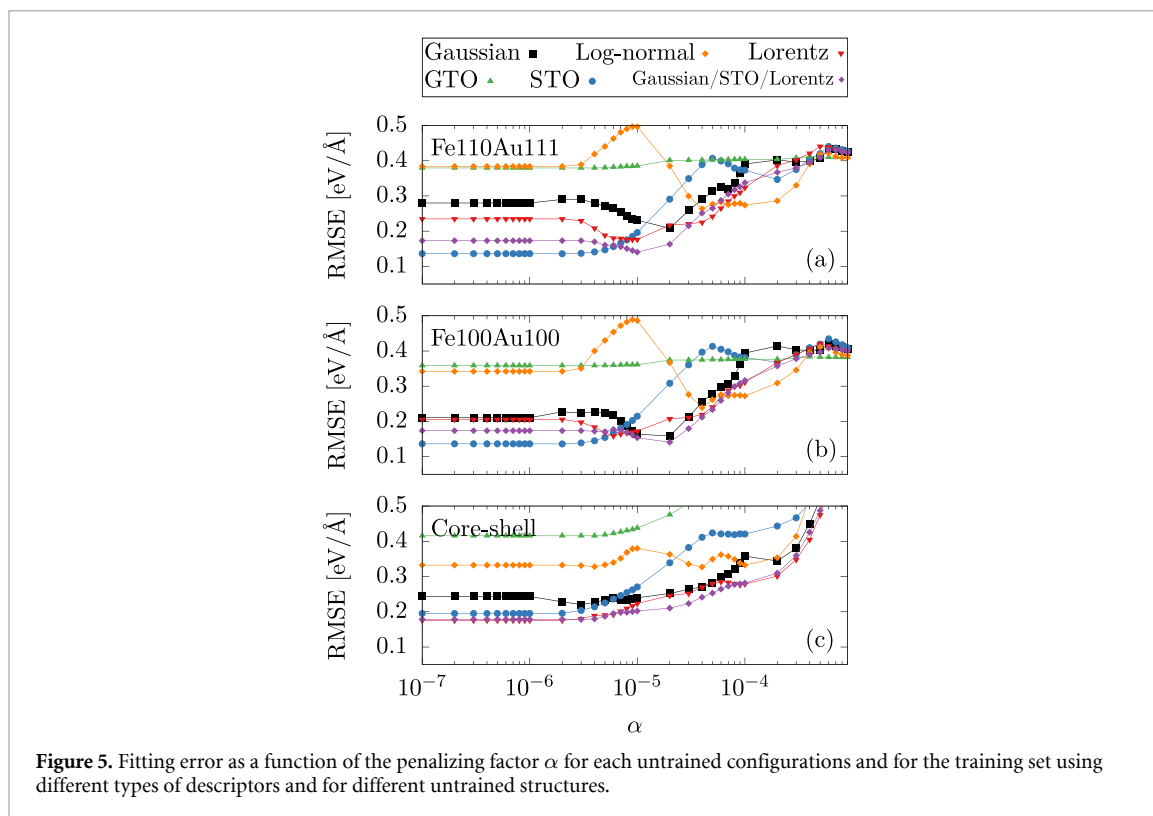


Figure 5. Fitting error as a function of the penalizing factor α for each untrained configurations and for the training set using different types of descriptors and for different untrained structures.

molecular dynamics software LAMMPS [63] in which the proposed MLIP was implemented [Please see appendix for details on the LAMMPS implementation]. Periodic boundary conditions were employed and the different minimization runs were performed down to a net force of 10^{-6} eV \AA^{-1} . We measured eight different lattice constants (pure iron bcc, pure gold fcc, alloys with 25%, 50% and 75% of gold in both bcc and fcc phases). In addition, our fitting did not include any energy. Therefore, to take into account the atomic energy, the MLIP energies are shifted in order to match the cohesive energy of pure iron and gold most stable states i.e. bcc and fcc respectively. Then, we also measured the cohesive energy for each alloying structures. Results are compared in figures 6(d) and (e) to DFT calculations that were previously obtained [47]. The errors are lower than 3% for lattice spacing and 12% for cohesive energy which is already satisfying considering that in the database, we employed alloying proportions that are much closer to 50% and only used nano-crystals. Therefore, being able to reach such small relative errors even for those extreme alloying proportions (25% and 75%) and in addition with bulk structures is an encouraging result for our MLIP.

Finally, we measured phonon dispersion curves using supercell approach implemented in PHONOPY [64] (see figures 6(e) and (f)). In practice, the dynamical matrix was obtained by moving each symmetrically independent atoms by 0.01 \AA . We used a supercell of size $4 \times 4 \times 4$. The agreement between DFT and MLIP curves is not as good as what is usually obtained in MLIP works [17, 65] but it remains qualitatively satisfying if one considers that our MLIP was not trained on any bulk structures.

Even if our potential has not been designed to reproduce bulk properties, these results are already very encouraging. Obtaining an accurate MLIP potential that is transferable to any phase and/or structure is a considerable challenge for multi-component systems and would require to carry out additional DFT calculations to build a bigger database including bulk structures but also interfaces. Besides, in order to target a specific application, one should also perform ‘on-the-fly’ optimization of the potential as proposed by Jinnouchi *et al* [39, 38]. However this is not the main purpose of this paper, which focuses on shedding light onto the relationship between complexity and transferability in machine learning force fields. It remains that our current MLIP potential may be used as a first step when studying bimetallic Fe–Au nanoparticles.

4. Discussion and conclusion

To summarize, this work aims at measuring transferability issues that can occur when using MLIP in untrained structures. To begin, we presented the employed MLIP method that includes a linearized potential and a penalizing regression scheme. Then, we discussed the influence of the descriptor space and showed that although the three different peak functions behave similarly in terms of accuracy and number of non-zero coefficients, the repulsive functions lead to a worst accuracy but with a lower number of non-zero

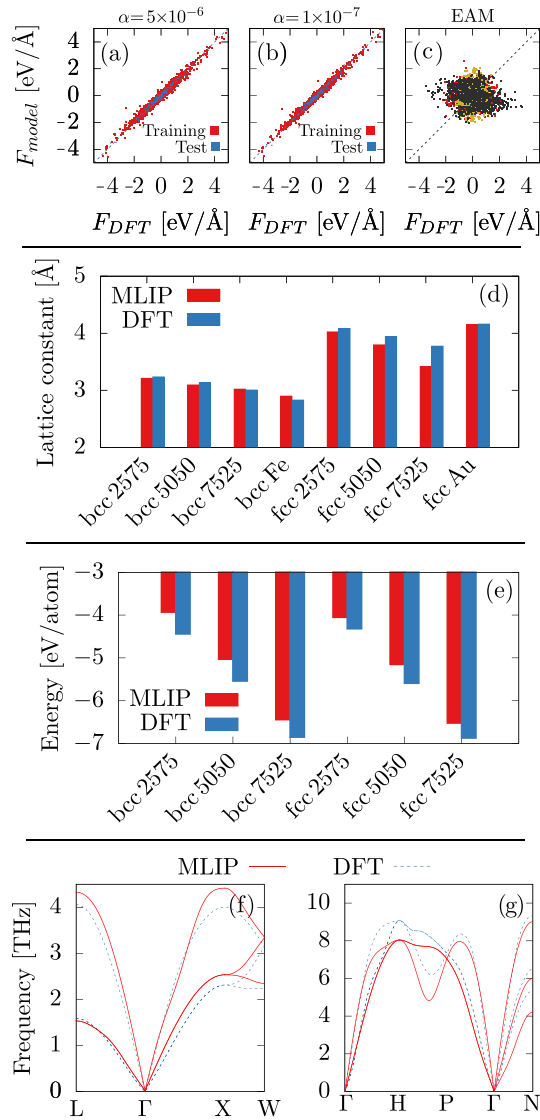
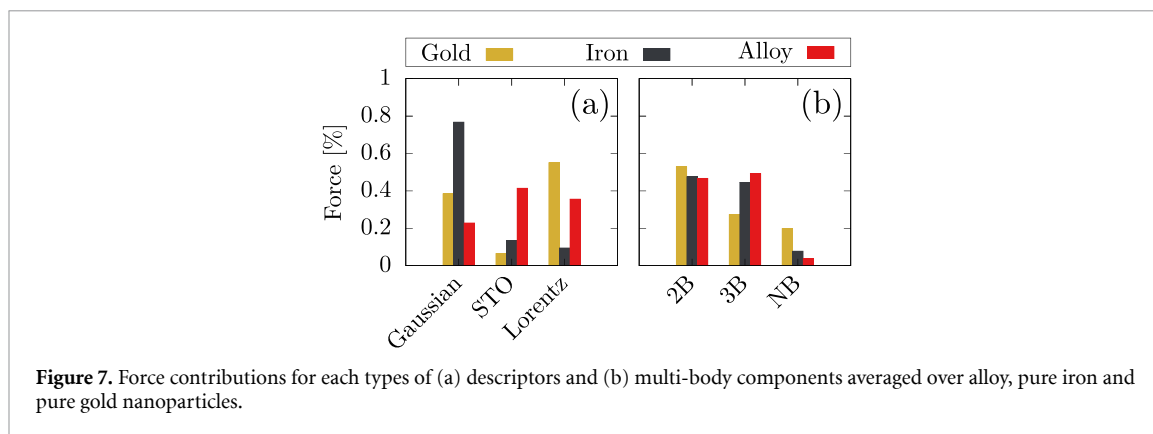


Figure 6. (a)–(c) Correlation plot between DFT forces and forces from MLIP with $\alpha = 10^{-7}$ (b) and with $\alpha = 5 \times 10^{-6}$ (c) and from EAM (d). Blue and red points correspond to results within the training and the test sets. (d,e) Values obtained for the lattice parameters and for the cohesive energies using MLIP and DFT. (f)–(g) Phonon dispersion of respectively gold FCC and iron BCC. The plain lines correspond to MLIP results while the dotted lines are obtained by DFT calculations.

coefficients. Next, we demonstrated that by using the LassoLars algorithm instead of the previously employed linear regression scheme, one can finely tune the complexity of the potential along with its accuracy. This is because the penalizing parameter α allows for turning off most of the initially proposed descriptors thus controlling the overall complexity of the potential. With this ability in hand, we measured transferability issues using three unlearned structures that are qualitatively different from what was considered in the training database. We showed that while the accuracy on the trained structures decreases monotonically as the value of α is decreased, it is not the case for those untrained structures. Indeed, when using only one type of descriptor, it exists an optimal value of α that allows for the best transferability. Finally, we introduced a way to overcome this transferability issue which consists in using different types of descriptors simultaneously. This again shows an other advantage of using the LassoLars algorithm which is able to actively select the most appropriate descriptors. Finally, we computed some properties of the Fe–Au in bulk and showed that the obtained potential is already qualitatively satisfying. But, before being able to really use our potential from practical applications, we plan to improve further it by adding bulk and interface DFT calculations within the database and by implementing ‘on the fly’ learning. As a perspective, the obtained MLIP will be further improved and then employed to study nucleation in core–shell FeAu nanoparticles as observed in experiments [41–45]. Before closing, we wish to discuss two additional points.

First, the very intuitive MLIP expression that was employed here allows us to give some insights on the nature of the interactions. Here, we focus on the combined MLIP where Gaussian, STO and Lorentzian peak



were used simultaneously. We can indeed distinguish between each descriptors (Gaussian, STO, Lorentzian). For that purpose, we compute the ratio between the absolute value of the forces given by each descriptor and the sum of the absolute values of the three descriptors and then perform an average over each force components (x, y, z) of all of the atoms within the training database (alloy, pure iron and pure gold nanoparticles). Figure 7(a) shows that the preponderant functions are different depending on the considered interactions thus highlighting the advantage of using simultaneously the three types of descriptors. More interestingly, the same can be done in order to distinguish between two-body, three-body and N-body contributions (see figure 7(b)). In our case, the two-body and three-body contributions are more important than the N-body contributions. This may explain why the previously employed EAM potentials that do not possess any explicit angular contributions could not accurately compute the forces.

Moreover, we would like to raise an additional implication of our work in which the transferability issues were measured and connected to the complexity of the potential. Indeed, as previously discussed, when using the LassoLars regression scheme, the complexity of the potential can be adjusted using the penalizing parameter. In the alternative MLIP methods, the same is done by modifying (1) the number of neurons and hidden layers in the case of neural-network potential [5] and (2) the number of selected configurations after sparsification in the case of gaussian approximation model [9, 32]. For some users of these techniques, the rule of thumb may be to use these adjusting parameters in order to increase the complexity of the potential which necessarily improves the accuracy on the learning database. Yet, our work indicates that transferability issues should be expected by such operation.

Data availability statement

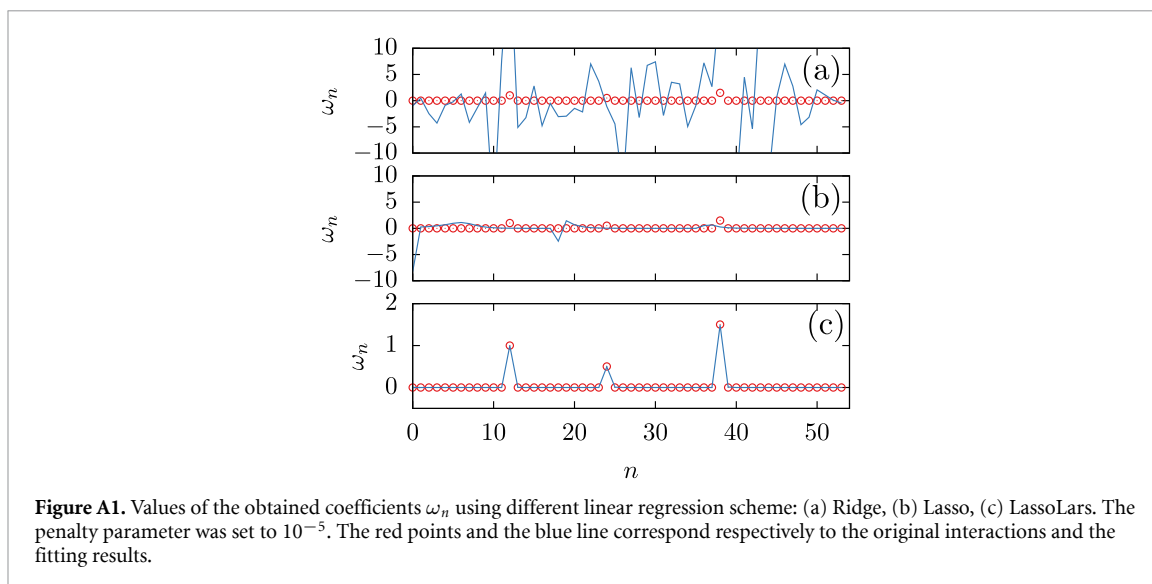
The data that support the findings of this study are available upon reasonable request from the authors.

Acknowledgments

JL acknowledges financial support of the Fonds de la Recherche Scientifique—FNRS. Computational resources have been provided by the Consortium des Equipements de Calcul Intensif (CECI), by the Fédération Lyonnaise de Modélisation et Sciences Numériques (FLMSN) and by the Regional Computer Center CALMIP in Toulouse (Grant No. p1141). JL thanks David Tew for having introduced him to the machine-learning potential problem and C. Patrick Royall for supervising the early method development. JL is also grateful to Francesco Turci, Joshua F Robinson and Atsuto Seko for fruitful discussions. Authors are finally grateful to Abdul-Rahman Allouche and Albert P Bartók for proofreading the manuscript.

Appendix A. Advantage of using LassoLars against other linear regression scheme

In this section, we wish to further motivate the choice of our regression scheme by working on a case study where we will compare LassoLars with two commonly employed linear regression schemes naming Ridge and the coordinate descent Lasso. For this test, the database is generated with an equimolar mixture of binary Lennard–Jones particles thus allowing us to directly verify the obtained MLIP. With such a binary system, an additional challenge for the fitting algorithm is to distinguish self-species and cross-species interactions. In practice, positions and forces were measured for 50 configurations of 64 atoms in the liquid regime. Regarding the basis of descriptors, only two-body interactions were considered and we used 17



Lennard–Jones functions with different distance parameters including those in the original simulations. All of the four employed methods manage to retrieve a linear combination of Lennard–Jones functions that matches the original interactions. However, it appears that only LassoLars can find the correct coefficients ω_n setting all coefficients to 0 except those of the original interactions (see figure A1). Such result shows the advantage of using LassoLars instead of the commonly employed linear regression methods.

Appendix B. Numerical implementation of the Φ -LassoLars method

In this section, we give some additional details on the described MLIP. First, obtaining an MLIP using the Φ -LassoLars method consists in two steps. In the first step, a homemade C++ parallelized code using OpenMP was developed to construct a matrix:

- Each columns of the matrix designates a specific descriptor which can be 2B, 3B or NB for all the functions and considered values for their parameters described in table 1.
- Each rows of the matrix correspond to the force on a given direction, atom and structure.

In the second step, a python code concatenates the obtained matrices for each structures and read the associated DFT forces. The same python code finally employs the LassoLars method as implemented in the `sklearn` package to obtain the linear coefficients associated to each columns of the matrix.

Then, in order to use the obtained MLIP, the same C++ code is employed to read the obtained coefficients and generate input files for LAMMPS simulations. Those consists on three parts:

- For the 2B interactions, we directly add all of the selected linear contributions and generate a table file that can be read by LAMMPS using `pair_style table`.
- For the 3B interactions, we build a homemade routine that is added to LAMMPS and use an personalized input file.
- For the NB interactions, we use a python code based on `atsim.potentials` [66] to generate EAM-like files that can be directly read by LAMMPS.

Finally, the `pair_style hybrid/overlay` is employed to combine all the contributions.

Regarding the computational timing for building the MLIP, a matrix for a structure of 64 atoms containing all of the functions (2B, 3B, NB for all values in table 1) for one type of descriptors is obtained in approximately 3 min. This process can be parallelized since each structure can be treated independently. Then, after reading all of the input matrices, a LassoLars fitting takes less than 20 s when using only one type of functions and less than 2 min when using simultaneously three types of functions. Results are obtained on one Intel E5-2650 processor.

ORCID iD

Julien Lam  <https://orcid.org/0000-0003-0343-6005>

References

- [1] Marx D 2009 *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods* (Leiden: Cambridge University Press)
- [2] Martin R 2008 *Electronic Structure: Basic Theory and Practical Methods* (Cambridge: Cambridge University Press)
- [3] Frenkel D and Smit B 2002 *Understanding Molecular Simulation* (New York: Elsevier)
- [4] Stone A J 2016 *The Theory of Intermolecular Forces* (Oxford: Oxford University Press)
- [5] Behler J 2015 *Int. J. Quantum Chem.* **115** 1032
- [6] Bartók A P and Csányi G 2015 *Int. J. Quantum Chem.* **115** 1051
- [7] Behler J 2016 *J. Chem. Phys.* **145** 170901
- [8] Behler J and Parrinello M 2007 *Phys. Rev. Lett.* **98** 146401
- [9] Bartók A P, Payne M C, Kondor R and Csányi G 2010 *Phys. Rev. Lett.* **104** 136403
- [10] Seko A, Takahashi A and Tanaka I 2014 *Phys. Rev. B* **90** 024101
- [11] Goryaeva A M, Maillet J-B and Marinica M-C 2019 *Comput. Mater. Sci.* **166** 200–209
- [12] Thompson A P, Swiler L P, Trott C R, Foiles S M and Tucker G J 2015 *J. Comput. Phys.* **285** 316
- [13] Chmiela S, Tkatchenko A, Sauceda H E, Poltavsky I, Schütt K T and Müller K-R 2017 *Sci. Adv.* **3** e1603015
- [14] Chmiela S, Sauceda H E, Müller K-R and Tkatchenko A 2018 *Nat. Commun.* **9** 1
- [15] Shapeev A V 2016 *Multiscale Model. Simul.* **14** 1153–73
- [16] Novoselov I I, Yanilkin A V, Shapeev A V and Podryabinkin E V 2019 *Comput. Mater. Sci.* **164** 46
- [17] Seko A, Takahashi A and Tanaka I 2015 *Phys. Rev. B* **92** 054113
- [18] Takahashi A, Seko A and Tanaka I 2018 *J. Chem. Phys.* **148** 234106
- [19] Zeni C, Rossi K, Glielmo A, Fekete A, Gaston N, Baletto F and De Vita A 2018 *J. Chem. Phys.* **148** 241739
- [20] Botu V, Batra R, Chapman J and Ramprasad R 2017 *J. Phys. Chem. C* **121** 511
- [21] Bereau T, DiStasio R A, Tkatchenko A and von Lilienfeld O A 2018 *J. Chem. Phys.* **DETC2018** 241706
- [22] Sauceda H E, Chmiela S, Poltavsky I, Müller K-R and Tkatchenko A 2019 *J. Chem. Phys.* **150** 114102
- [23] Bartók A P, De S, Poelking C, Bernstein N, Kermode J R, Csányi G and Ceriotti M 2017 *Sci. Adv.* **3** e1701816
- [24] Veit M, Jain S K, Bonakala S, Rudra I, Hohl D and Csányi G 2019 *J. Chem. Theory Comput.* **15** 2574
- [25] Artrith N, Morawietz T and Behler J 2011 *Phys. Rev. B* **83** 153101
- [26] Quaranta V, Hellström M and Behler J 2017 *J. Phys. Chem. Lett.* **8** 1476
- [27] Nguyen T T, Székely E, Imbalzano G, Behler J, Csányi G, Ceriotti M, Götz A W and Paesani F 2018 *J. Chem. Phys.* **DETC2018** 241725
- [28] Bartók A P, Gillan M J, Manby F R and Csányi G 2013 *Phys. Rev. B* **88** 054104
- [29] Morawietz T, Sharma V and Behler J 2012 *J. Chem. Phys.* **136** 064103
- [30] Natarajan S K and Behler J 2016 *Phys. Chem. Chem. Phys.* **18** 28704
- [31] Morawietz T and Behler J 2013 *J. Phys. Chem. A* **117** 7356
- [32] Deringer V L and Csányi G 2017 *Phys. Rev. B* **95** 094203
- [33] Bartók A P, Kermode J, Bernstein N and Csányi G 2018 *Phys. Rev. X* **8** 041048
- [34] Caro M A, Deringer V L, Koskinen J, Laurila T and Csányi G 2018 *Phys. Rev. Lett.* **120** 166101
- [35] Deringer V L, Bernstein N, Bartók A P, Cliffe M J, Kerber R N, Marbella L E, Grey C P, Elliott S R and Csányi G 2018 *J. Phys. Chem. Lett.* **9** 2879
- [36] Deringer V L, Caro M A, Jana R, Aarva A, Elliott S R, Laurila T, Csányi G and Pastewka L 2018 *Chem. Mater.* **30** 7438
- [37] Sosso G C, Deringer V L, Elliott S R and Csányi G 2018 *Mol. Simul.* **44** 866
- [38] Jinnouchi R, Lahnsteiner J, Karsai F, Kresse G and Bokdam M 2019 *Phys. Rev. Lett.* **122** 225701
- [39] Jinnouchi R, Karsai F and Kresse G 2019 *Phys. Rev. B* **100** 014105
- [40] Li Z, Kermode J R and De Vita A 2015 *Phys. Rev. Lett.* **114** 096405
- [41] Langlois C, Benzo P, Arenal R, Benoit M, Nicolai J, Combe N, Ponchet A and Casanove M J 2015 *Nano Lett.* **15** 5075
- [42] Benzo P, Combettes S, Pecassou B, Combe N, Benoit M, Respaud M and Casanove M J 2019 *Phys. Rev. Mater.* **3** 096001
- [43] Tymoczko A, Kamp M, Rehbock C, Kienle L, Cattaruzza E, Barcikowski S and Amendola V 2019 *Nanoscale Horiz.* **4** 1326
- [44] Tymoczko A, Kamp M, Prymak O, Rehbock C, Jakobi J, Schürmann U, Kienle L and Barcikowski S 2018 *Nanoscale* **10** 16434
- [45] Ponchet A, Combettes S, Benzo P, Tarrat N, Casanove M J and Benoit M 2020 *J. Appl. Phys.* **128** 055307
- [46] Vernieres J, Steinhauer S, Zhao J, Grammatikopoulos P, Ferrando R, Nordlund K, Djurabekova F and Sowwan M 2019 *Adv. Sci.* **6** 1900447
- [47] Calvo F, Combe N, Morillo J and Benoit M 2017 *J. Phys. Chem. C* **121** 4680
- [48] Hong S and Rahman T S 2015 *Phys. Chem. Chem. Phys.* **17** 28177
- [49] Combettes S, Lam J, Benzo P, Ponchet A, Casanove M-J, Calvo F and Benoit M 2020 *Nanoscale* **12** 18079
- [50] Zhou X W, Johnson R A and Wadley H N G 2004 *Phys. Rev. B* **69** 144113
- [51] Takahashi A, Seko A and Tanaka I 2017 *Phys. Rev. Mater.* **1** 063801
- [52] Seko A, Togo A and Tanaka I 2019 *Phys. Rev. B* **99** 214108
- [53] Wood M A and Thompson A P 2018 *J. Chem. Phys.* **DETC2018** 241721
- [54] Wang H, Guo X, Zhang L, Wang H and Xue J 2019 *Appl. Phys. Lett.* **114** 244101
- [55] Stillinger F H and Weber T A 1985 *Phys. Rev. B* **31** 5262
- [56] Daw M S and Baskes M I 1984 *Phys. Rev. B* **29** 6443
- [57] Efron B, Hastie T, Johnstone I and Tibshirani R 2004 *Ann. Stat.* **32** 407
- [58] Hung N, Hue T and Duc N 2015 *VNU J. Sci.: Math. Phys.* **31** (<https://js.vnu.edu.vn/MaP/article/view/114>)
- [59] Kresse G and Furthmüller J 1996 *Comput. Mater. Sci.* **6** 15
- [60] Blöchl P E 1994 *Phys. Rev. B* **50** 17953
- [61] Zuo Y et al 2020 *J. Phys. Chem. A* **124** 731
- [62] Benoit M, Langlois C, Combe N, Tang H and Casanove M-J 2012 *Phys. Rev. B* **86** 075460
- [63] Plimpton S 1995 *J. Comput. Phys.* **117** 1
- [64] Togo A and Tanaka I 2015 *Scr. Mater.* **108** 1
- [65] Byggmästar J, Hamedani A, Nordlund K and Djurabekova F 2019 *Phys. Rev. B* **100** 144105
- [66] Rushton M atsim.potentials—Potential Model Tabulation for Atomic Scale Simulation 2020 (<https://atsimpotentials.readthedocs.io/>)