Comparing transferability in neural network approaches and linear models for machine-learning interaction potentials

Akshay Krishna Ammothum Kandy,[1] Kevin Rossi,[2, 3] Alexis Raulin-Foissac,[1] Gaétan Laurens,[4] and Julien Lam[1, *]

[1] *Centre d'élaboration des Matériaux et d'Etudes Structurales,*
CNRS (UPR 8011), 29 rue Jeanne Marvig, 31055 Toulouse Cedex 4, France
[2] *Institut des sciences et ingénierie chimiques, École Polytechnique Fédérale de Lausanne, 1950 Sion, Switzerland*
[3] *Institute for Chemical and Bio-engineering, Department of Chemistry and Applied Biosciences,*
ETH Zurich, Vladimir-Prelog-Weg 1, 8093 Zurich, Switzerland
[4] *Institut Lumière Matière, UMR5306 Université Lyon 1-CNRS,*
Université de Lyon, 69622 Villeurbanne Cedex, France

Atomic simulations using machine learning interatomic potential (MLIP) have gained a lot of popularity owing to their accuracy in comparison to conventional empirical potentials. However, the transferability of MLIP to systems outside the training-set poses a significant challenge. Here, we compare the transferability of three MLIP approaches: i) Neural Network Potentials (NNP), ii) Physical LassoLars Interactions Potential (PLIP) and iii) Linear Potentials with Belher-Parrinello descriptors, trained over a small but diverse configuration of zinc oxide polymorphs. We compared the obtained models with density functional theory reference results for physical properties including bulk lattice parameters, surface energies, and vibrational density of states and showed the superiority of both NNP and PLIP models. However, the NNP model performed poorly when compared to the other two linear models for the structural optimization of nanoparticles and molecular dynamics simulation of liquid phases, which are systems outside the training-set. While providing less accurate prediction for solid Zinc Oxides phases, both linear models appear more transferable than NNP when testing for nanoscale systems and liquid phases. Our results are finally rationalized by a combination of different statistical analysis including spread in force evaluation, information imbalance, convex hull calculation and density in descriptor space.

I. INTRODUCTION

Atomistic simulations have played a crucial role in the discovery of novel materials and the understanding of their specific properties. In this context, quantum calculations, including ab initio and density functional theory (DFT), are the most accurate for calculating equilibrium properties and provide quantitative results comparable to experiments. However, even with the current technical progress, computational time prevents us from performing quantum accurate simulations that involve more than a thousand atoms at nanosecond timescales. In the meantime, empirical interaction potentials that are constructed to match material properties measured experimentally or computed with quantum calculations have been employed to perform such large-scale simulations. Yet, the use of empirical interaction potential yields significantly lower accuracy, when compared to quantum calculations. Machine-learning interaction potentials (MLIP) have been recently proposed to bridge the gap between quantum accurate calculations and fast empirical modeling [1, 2]. The main principle consists in using a large set of quantum-accurate calculations to adjust the parameters of a universal mathematical formulation that should represent the interaction potential. Lots of different approaches have been proposed, including Artificial Neural Networks[3], Gaussian approximation potentials[4], Linearized potentials[5–10], Spectral Neighbor Analysis Potential[11, 12], Symmetric Gradient Domain Machine learning[13, 14], Moment Tensor Potentials[15, 16], Atomic Cluster Expansion,[17–19] and E(3)-equivariant interatomic potentials[20]. Meanwhile, lots of different materials have also been successfully modeled with those machine-learning interaction potentials (MLIP) including pure metals[5, 21–24], organic molecules[25–28], water[29–33], amorphous materials[34–39], and hybrid perovskites[40, 41].

In practice, a key aspect of the machine-learning approach is the intrinsic relationship between the learning database and the scope of application for the obtained potential. In particular, while it is clear that robust results can be expected when sampling structures that are within the reach of the learning database, applying the obtained potential in out of training distribution is much more problematic. In this context, it appears crucial to multiply studies of transferability issues which may render the practical usage of MLIP difficult [10, 42]. In particular, it is generally mentioned that an overly complex formulation of the potential, in terms of the employed descriptors and the embedding functions, may render the obtained model less transferable.[10] As such, neural network potentials and linear models are often confronted as they are located at both sides of the complexity spectrum.

In this article, we worked on the example of zinc oxide interactions and first constructed three types of machine-learning interaction potentials using Neural Network Potential (NNP), Linear Behler-Parrinello potential (LBP) and the Physical LassoLars Interactions

[*] julien.lam@cnrs.fr

Potential (PLIP). Then, we compared the root-mean-square error on both learning and testing databases. Next, the three models are employed to measure several physical properties of zinc oxide including lattice parameters, surface energies, and vibrational density of states (vDOS). Finally, we designed two stringent transferability tests for NNP, LBP, and PLIP: (1) Molecular Dynamics (MD) simulation of liquids, and (2) Structure optimization of nanoparticles. Altogether, our results suggest that, although NNP generally performs better than both PLIP and LBP in learned situations, its applicability range can be lower for out-of-distribution structures. The obtained result is finally analyzed using statistical metrics including spread in force prediction, information imbalance, convex hull inclusion and sampling density. This work thus contributes to a better understanding of the possibilities and limitations of machine-learning interaction potentials, especially for the approaches including linear and feed-forward neural networks models.

II. METHODS

A. Database

The training dataset was built in a previous work [43] by means of first-principle calculations. Six ZnO polymorphs, *i.e.* wurtzite (WRZ), zinc blend (ZBL), body-centered tetragonal (BCT), sodalite (SOD), *h*-BN (HBN), and cubane (CUB) crystallographic structures were considered in the database. We first employed MD simulations using classical Buckingham ZnO potential[44] to melt of the 6 crystal structures up to 5000 K. We extracted 20 snapshots per crystal polymorph and computed forces using DFT calculations. With this first database, we constructed a first PLIP model that was used to perform MD simulations of the melting from surface structures up to 2000 K. In particular, we used as a starting point, 7 different surfaces. Along the melting path, 50 snapshots were collected and used in the database after having computed the forces with DFT calculations. Finally, we constructed a second PLIP model with this supplemented database. We performed MD simulations to obtain amorphous structures by rapid temperature quench of the liquid configurations that were previously obtained. We extracted 58 additional structures that were used as input for force calculations in DFT. Overall, machine-learning models were trained on a database with a size of 87699 atomic environments. We note that although it should not considerably affect the obtained results, some structures of the database were generated with the PLIP model.

First-principles calculations were insured using the GGA-PW91 exchange-correlation functional [45], and the projector augmented wave method [46] implemented in VASP [47, 48]. More details are provided in Ref. [43].

B. Machine-learning models

Construction of machine-learning interaction potentials (MLIP) was achieved using NNP, LBP and PLIP. In all approaches, the total potential energy E is approximated as the sum of independent atomic energies E_i: $E = \sum_{i=0}^{N} E_i$, where N corresponds to the number of atoms of the considered configuration.

Firstly, the PLIP model consists in approximating E_i as a linear combination of descriptors X_n^i:

$$E_i = \sum_n \omega_n X_n^i \tag{1}$$

where ω_n are the linear coefficients that must be determined. We considered three types of body-ordered descriptors:

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

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

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

where r_{ij} is the distance between atoms i and j, θ_{ijk} is the angle centered around the atom i, and l (≤ 5) and m (≤ 7) are two positive integers. The cutoff function is defined as: $f_c(r_{ij}) = 0.5\,(1 + cos(\pi(r_{ij}/r_{cut})))$, with r_{cut} is set at 6 Å. The shift function takes the form: $f_s(u) = 6u^5 - 15u^4 + 10u^3$, where $u = (r_{ij} - r_1)/(r_2 - r_1)$, and r_1 (resp. r_2) is defined as 95 % (resp. 105 %) of a distance equal to 1.1 Å. Regarding the basis functions $f_n(r_{ij})$, we used Gaussian functions centered around values between [0.5, 1.0,..., 5.5, 6.0], and with widths ranged within [0.5, 1.0, 1.5]. Altogether, our model is made of 1980 available descriptors. For the fitting of the DFT database, the LassoLars approach is employed to perform a well-informed selection of the most preponderant descriptors and a reduction in the complexity of the obtained potential [10]. In particular, we used a penalizing factor equal to 10^{-5} and the final number of descriptors after sparsification are 94, 96, 95, 98 and 91 out of the 1980 available descriptors.

Secondly, we constructed NNP models following the Behler and Parrinello approach [3] and using the neural network potential package, *n2p2* [49, 50]. The atomic energy $E_i(G_j)$ depends on the descriptors defined as the symmetry functions G_j:

$$G_j^2 = \sum_{j\neq i}^{N} e^{-\eta(r_{ij}-r_s)^2} f_c(r_{ij}) \tag{5}$$

$$G_j^3 = 2^{1-\zeta} \sum_{j,k\neq j}^{N} (1+\lambda\cos(\theta_{ijk}))^{\zeta} e^{-\eta(r_{ij}^2+r_{ik}^2+r_{jk}^2)} \tag{6}$$

$$f_c(r_{ij}) f_c(r_{ik}) f_c(r_{jk})$$

$$G_j^9 = 2^{1-\zeta} \sum_{j,k\neq j}^{N} (1+\lambda\cos(\theta_{ijk}))^{\zeta} e^{-\eta(r_{ij}^2+r_{ik}^2)} \tag{7}$$

$$f_c(r_{ij}) f_c(r_{ik}))$$

The cutoff function was chosen as polynomial: $f_c(x) = 1+((15-6x)x-10)x^3$ with $x = r_{ij}/r_{cut}$. The free parameters of the two-body symmetry functions, η and r_s, are set at 2 Å^{-2} and [1.5, 2.0,..., 5.0, 5.5] Å, respectively. For the three-body symmetry functions, values of [-1;1] are used for λ, [1;6] for ζ, and [0.2222; 0.004082; 0.01653] Å^{-2} for η. These descriptors are injected into the input layer of the NNP which is composed of two hidden layers of 15 nodes connected by the soft-plus activation function implemented in the $n2p2$ package. We used the same set of parameters that were already considered when constructing NNP using *n2p2* [50]. For the learning process, we considered the extended Kalman filter, which shows faster convergence than stochastic gradient descent and the Adam method [50]. The fitting is stopped after 150 epochs and we choose the best epoch as the one minimizing the product of root-mean-square errors for training and testing sets.

Thirdly, we constructed LBP models using the same descriptors as in the NNP models. Yet, instead of having a feed-forward neural network, the descriptors are directly considered in a linear model as in Eqn. 1. For practical reasons, the learning process was also carried out by *n2p2* with the same protocol as described previously.

To compute the error bar in our measurements, we constructed 5 different MLIPs with each approach. In all cases, it meant changing the randomly selected training database (90%). In the case of NNP, we also modified the initial weights of the neural network architectures. We note that in the three cases, the fitting is performed solely with forces thus avoiding the addition of a hyperparameter that would scale force and energy contributions. The three models were implemented in LAMMPS which is used to perform all of the following calculations [51, 52].

After describing the models and the corresponding training process, let us comment on the differences between each models. In particular, the PLIP descriptors are the same as in our previous works on gold/iron[10] and on the ZnO,[43] and are derived from the work of Seko et al.[5, 6, 8]. Through Eqns. (2-4) we decompose the atomic energy as contributions arising from two, three and N body interactions. We note that the N-body interactions as put forward in Eqn. 4 are a generalization of the classical empirical model of EAM potentials [See Ref. [7] for mathematical demonstration]. Moreover, because Gaussian functions are used for $f_n(x)$, the PLIP descriptors are very similar to BP descriptors but three differences can still be noted. Firstly, the set of BP descriptors lack explicit n-body descriptors which is compensated by the neural network non-linear architecture. Secondly, the PLIP 3-body descriptors are not explicitly taking into account the three distances in the Gaussian functions and only use r_{ij} and r_{ik} as well as the cosine of the formed angle. Thirdly, because the LassoLars regression is used for descriptor selection, more values of central position and width can be put forward as initial descriptors. To finish, we note that a key component when comparing the three models is their inherent complexity. By using the number of fitting parameters, the order of complexity is NN, PLIP and LBP with numbers respectively equal to 3274, 1980 186.

III. RESULTS

A. Validation and benchmark on bulk training configurations

1. *Fitting errors*

In Fig. 1, we show that NNP can reach values of root-mean-square error (RMSE) that are almost two times better than PLIP for both training and testing sets. This is a major achievement for the NNP model which takes advantage of the higher flexibility of the mathematical formulation. To corroborate this hypothesis, the linear model obtained with the Behler-Parrinello descriptors leads to errors that are 4 and 3 times larger than the one of NNP and PLIP models.

Before testing the accuracy of the obtained MLIP for different physical properties, we also wish to verify the influence of the database size. For that purpose, we constructed five MLIPs for different fractions of the database and measured the corresponding RMSE using a separated testing database. Fig. 1.b shows that albeit being the less accurate model, LPB is the most stable when decreasing the amount of data used for training and that both NNP and PLIP begin suffering overfitting issues when using only 10% and 5% of the database. At this stage, it seems that the observed increase of PLIP error seems to slightly favor the usage of NNP in the regime of small datasets. Altogether, this justifies that we used models including up to the 90% of the database.

After having evaluated the fitting performances of both methods, we will now measure physical properties related to the zinc oxide materials using both MLIP models.

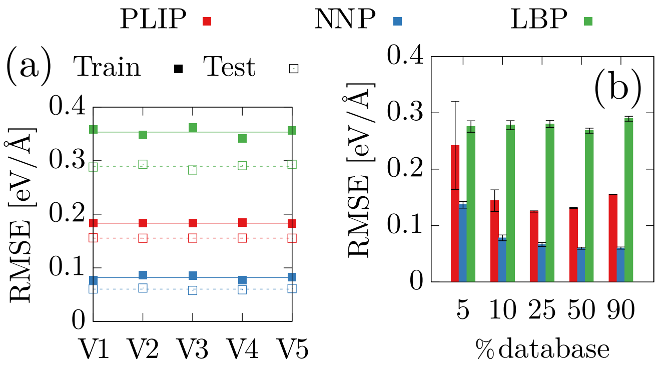


FIG. 1. (a) Root-mean-square errors on forces for five different trials using 90% of the entire database. Plain and dotted lines correspond respectively to the average training and testing errors. (b) Influence of the database size averaged over five trials.

2. *Lattice spacing*

The bulk lattice parameter a of each polymorph was computed from energy and force minimization and compared to DFT measurements. Fig. 2 displays the errors of the lattice parameter for the three considered models. Overall, they all lead to good results, within a relative error respect to the ab initio truth value, below 1 %, except for HBN which was poorly reproduced by LBP. In general, this physical quantity is better predicted by NNP in all of the crystal polymorphs. The NNP accuracy is especially significant for the energetically most stable polymorphs, *i.e.*, the wurtzite and the zinc blend structures. At this stage, NNP should be favored by comparison with both PLIP and LBP whose results remain sufficient for most practical usage.

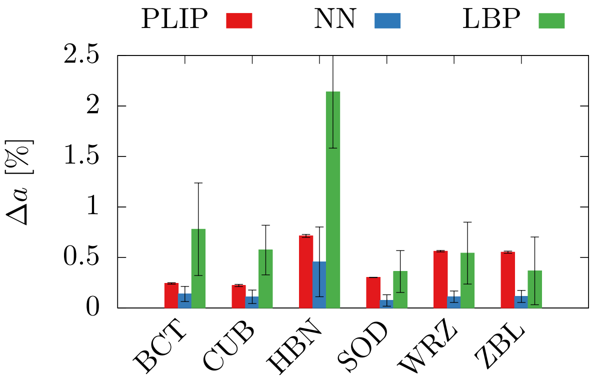


FIG. 2. Evaluation of the errors Δa made by the PLIP and the NN models for the prediction of the lattice parameter of each studied ZnO polymorph relative to the DFT reference values.

3. *Surface energy*

We will now measure errors made on surface energies denoted γ. For each crystal polymorph, the most stable surface has been considered, as well as some additional surfaces for the BCT and WRZ phases. From a quantitative perspective, errors on γ are larger than what was obtained for lattice spacing as they reach values of tens of percent, as shown in Fig. 3. When comparing the three methods, NNP again generally provides better results for the studied surfaces with LBP being the worst model.

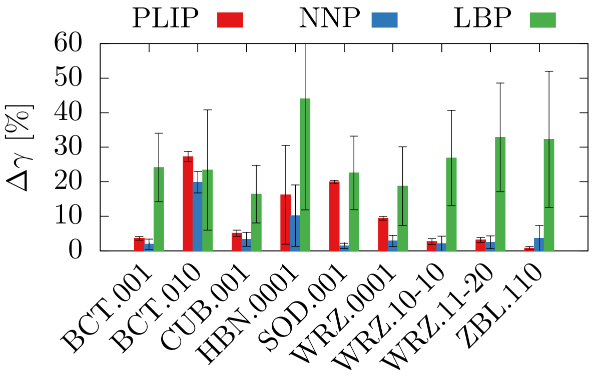


FIG. 3. DFT-relative errors on the surface energies of one or several surfaces for each ZnO polymorph computed by the PLIP and the NN models.

4. *Phonon density of states*

We further investigated the validity and accuracy of the MLIPs by performing lattice dynamics calculations. For this purpose, we considered WRZ, BCT, and ZBL bulk ZnO polymorphs. The phonon calculations were performed using a 5 × 5 × 5 supercell and the Phonopy package[53]. Results comparing phonon density of states (DOS) between DFT and MLIPs are shown in Fig. 4. Both the NNP and PLIP models can reproduce the DFT phonon density of states for all the ZnO polymorphs with reasonable accuracy. We note that relatively larger discrepancies are seen for DOS at higher frequencies (optical modes) than for the lower frequencies. In general, it remains that NNP leads to results that are again slightly better than PLIP. In this case, results of LBP are really unsatisfying when compared to both NNP and PLIP.

Altogether, measurements of physical properties that are directly related to structures within the training database allow us to conclude that LBP is really the worst model and that NNP performs slightly better than PLIP. It remains that both NNP and PLIP still exhibit very good results.

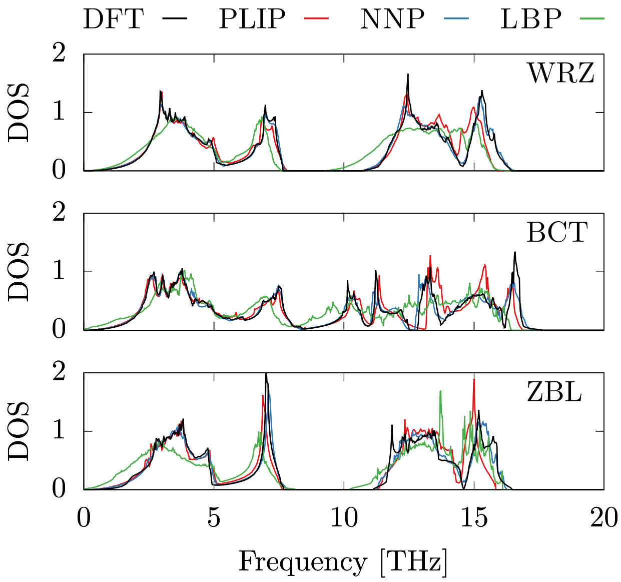


FIG. 4. Comparison of vibrational density of states (DOS) between DFT and the MLIP models for wurtzite (WRZ), body-centered tetragonal (BCT), and zincblende(ZBL) ZnO polymorphs.

B. Transferability towards other phases

1. *Liquid radial distribution*

To investigate the transferability of MLIPs, beyond the training data set configurations, we performed MD simulation of liquid ZnO. The system was initialized with 250 atoms structures obtained by DFT minimization of an amorphous structure. Then, the system temperature was increased from 300 K to 1500 K during 1 ps before equilibration at the same temperature during 1 ps as well. In all cases, we worked in the NVT ensemble with a Nose-Hoover thermostat. The MD simulations were performed using the five different PLIP, LBP and NNP models. The corresponding partial radial distribution functions (RDF) are shown in Fig. 5. The RDFs obtained using PLIP (solid red line) are in good agreement with the DFT reference (solid black line) and all five of them lead to very similar results. The LBP model behaves similarly to the PLIP model with slightly worst results for the ZnO distribution at the end the first peak. In the meantime, the NNP RDFs (blue lines) display spurious peaks, especially, at non-physical shorter distances. Moreover, it appears that only 3 of the 5 constructed NNPs were able to initialize simulations at 300 K although it was started with an amorphous structure obtained in DFT. The others would directly lead to enormous forces, which would remove atoms from the simulation box. Furthermore, among the remaining three, only one did not lead to spurious peaks at short ranges. As such, the studied case shows that although NNPs are powerful in the interpolation of complex data, they can also lead to non-physical behaviour in the extrapolation regime. In this case, the issue with NNPs is its inability to reproduce short range repulsion that could prevent from having those non-physical peaks at short distances.

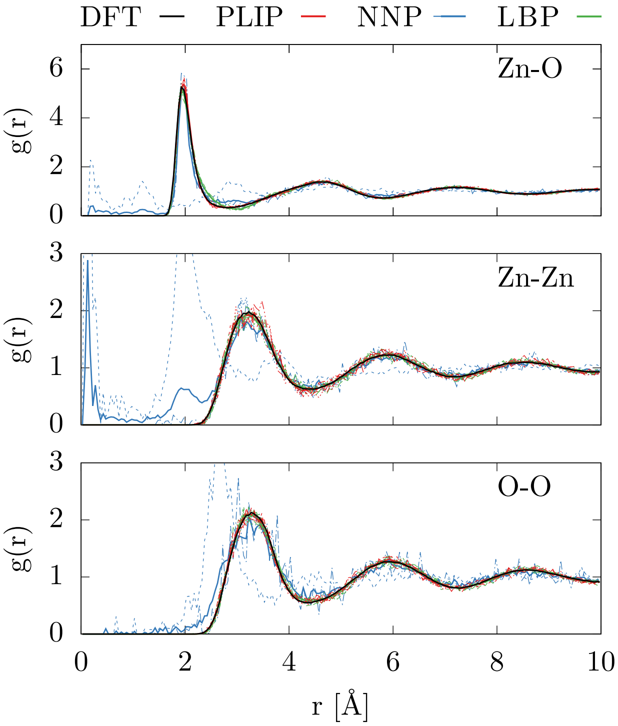


FIG. 5. The comparison for partial radial distribution functions $g(r)$ for structures obtained at 1500 K. The DFT reference is shown in solid black lines, PLIP, NNP and LBP models are respectively shown in red, blue and green lines. Each individual model is plotted with a dashed line while the solid line is the average over all the models. There are 5 dashed lines for both PLIP and LBP but only 3 dashed lines for NNP.

2. *Transferability towards nanoparticles*

To further test the extent of transferability of MLIPs, we used the obtained models to optimize ZnO nanostructures. Since the MLIPs were trained on periodic systems, nanostructures represent a significant transferability test. The MLIPs were therefore tested on nanoparticles made from BCT, SOD, and WRZ crystal phases, as well as single-caged (SC) and multi-caged (MC) structures. In particular, the calculations were initialized using DFT optimized structures as obtained by Vines et al.[54] with a slightly different DFT framework. Then, further DFT calculations were performed to reach the optimal configurations within our own DFT framework. Next, we randomly disturbed the atomic positions of the obtained structures by at most 0.3 Å. This step is made with three different random seeds. Finally, disturbed structures are optimized using the 5 MLIPs of each models. In total, for the three models, we can therefore average over 15 different structures for each nanoparticle. To quantify the error, the mean square deviation of atoms between MLIP optimized nanostructures and DFT optimized nanostructures was measured and averaged over 5 five different versions of the MLIP and 3 different random displacements. From Fig. 6, we can see that errors obtained are remarkably lower with PLIP and with LBP than with NNP for all of the considered nanoparticles. We also note that again the error bars are higher with NNP models.

In order to better understand the results in terms of the physics behind the observations, we isolated 5 typical nanoparticles, measured the coordination number using a cutoff equal to 2.5 Åand computed the MSD error for each type of coordination number [See Fig. 7(a_1, b_1, c_1, d_1, e_1]. First, we note that similar to what we previously observed when averaging over all of the coordination numbers, PLIP leads again to better results than both NNP and LBP. Then, it appears that the lowest coordination number are giving the highest MSD. In the meantime, the corresponding images show that atoms on the edges of the nanoparticles are less well reproduced. We speculate that the difficulty of NNP and LBP to reproduce the atoms located on the edges stem from the fact that those types of configurations were not inside the database as they are specific to nanoparticles.

IV. DISCUSSION

Until now, we showed that NNP provide slightly better results during the validation stage. However, supplementary tests regarding liquid radial distributions as well as nanoparticle structure minimization demonstrated that the two linear models (PLIP and LBP) provide more consistent results in domains which are characterized by physics and material chemistry different from the training one.

As a first step in understanding this result, we studied, for each approach, the spread in the force predictions between each of the obtained five potentials. As illustrated in Fig.8.a, the standard deviation in model predictions in train, test and nanoparticle points shows a different behavior than the RMSE in forces. In particular, while NNP provide the overall most accurate predictions in training, the spread in prediction is large. For PLIP instead we note a better agreement between each of five potentials. This result parallels the bigger error bars that were obtained when measuring physical properties in the bulk phases. Furthermore, this trend is heightened when performing inference in nanoscale structures. This result is thus a first evidence that nanoscale configurations appear in a domain of lesser robust extrapolation for the NNP.

As an additional route to rationalize the observed transferability behavior, we analyze the information encoded in the descriptor space employed in the build up of the MLIP. For that purpose, we considered respectively the entire set of PLIP descriptors, the latent space after neural network convolution and the entire set of BP descriptors.

To compare the information encoded in each of the models, we evaluate the information imbalance using the methodology introduced by Glielmo et al. [55]. In particular, given any two metrics A and B, this analysis probes whether: (1) the two are equivalent ($\Delta A \rightarrow B \sim 1$, $\Delta B \rightarrow A \sim 1$), (2) orthogonal ($\Delta A \rightarrow B \sim 0$, $\Delta B \rightarrow A \sim 0$), (3) if the information of one is contained also in the other ($\Delta A \rightarrow B \sim 0$, $\Delta B \rightarrow A \sim 1$), or (4) if the two offer equivalent but independent information ($0 << \Delta A \rightarrow B = \Delta B \rightarrow A << 1$). We report the information imbalance between the three considered representations in Fig. 8.b. The BP features are rather independent, if not orthogonal, to the two other representations. This finding hold both for the training and nanoscale configurations. This measure agrees well with the strong difference in accuracy between the LBP model and the other two. The PLIP and NNP representations carry information which is largely equivalent and independent. Consequently, we would expect similar, yet not identical trends for models which leverage either the former or the latter. The PLIP information is often larger than the one of the BP descriptors, and this is rationalized in terms of the larger number of features (sim 90 vs 15).

To extend our assessment of the transferability (or lack thereof) of the proposed ML models, we constructed the convex hull associated to all the points in the training set. Then, we tested if each data points belong to the convex hull [See Fig. 8.c]. We note that when doing so for the training data points, the convex hull is reconstructed by removing the considered training data point. In the case of the LBP, we observe that all the atomic environments appear outside the convex hull. The fraction of points within the convex hull is also small when considering the NNP representation with training, testing and nanoparticle data points being less and less present in the convex hull. The PLIP representation also shows a decrease when going from training to testing and nanoparticles. However, the numbers remain always much larger than both in LBP and NNP.

Finally, we recently demonstrated that the fraction of points inside the convex hull is not sufficient to address the accuracy of a machine-learning model[42]. As such, to complement the observations related to the convex hull, we measure the sampling density induced by the training point on the test points that is defined as:

$$\rho(\mathbf{x}^*) = \frac{k^* - 1}{M\, V^*}, \tag{8}$$

where M labels the training set site and V^* corresponds to the volume enclosing the first k^* training point neighbors [42, 56]. In addition, k^* is chosen adaptively, according to the framework described by Rodriguez et al.[57], and V^* corresponds to a d-dimensional hypersphere volume, where d corresponds to the intrinsic dimension of the training data manifold, estimated via the TwoNN

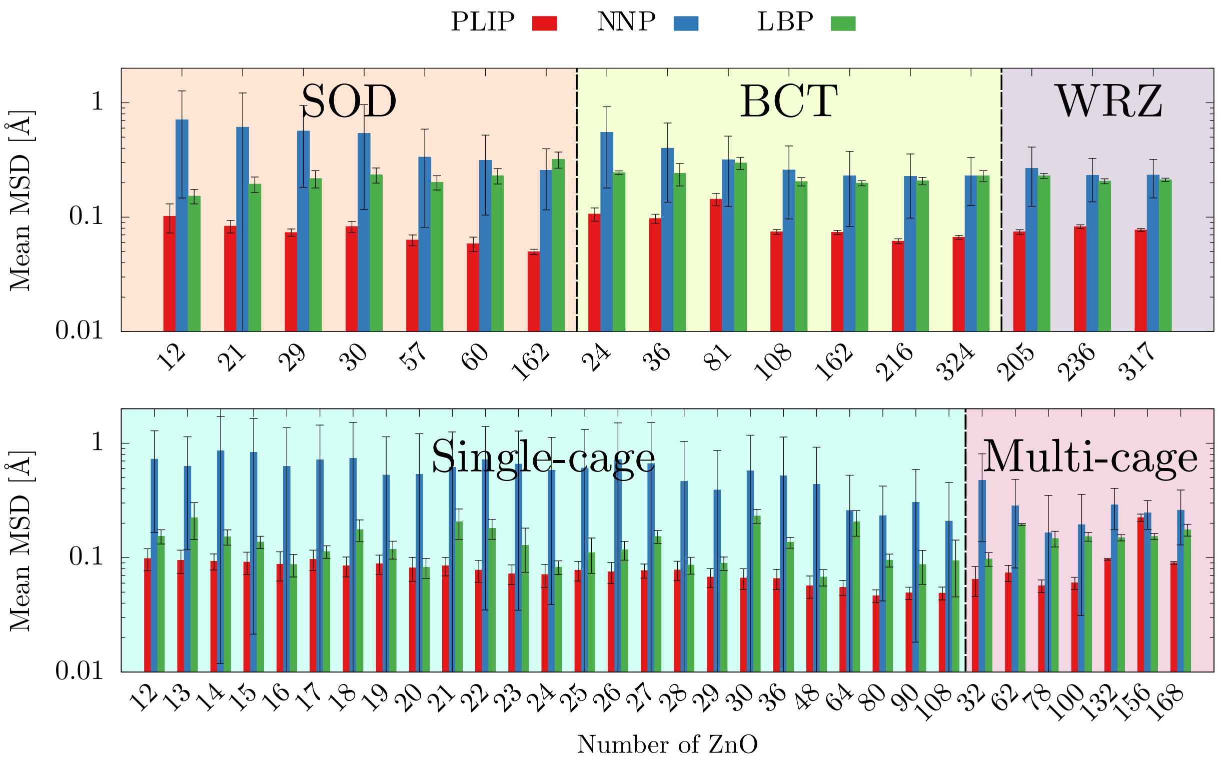


FIG. 6. Errors made by the two ML models on clusters energies relatively to those obtained by DFT. The BCT, SOD, and WRZ polymorphs are considered as well as multi-caged (MC) and single-caged (SC) clusters, provided from the work of Ref [54].

approach by Facco et al.Facco *et al.* [58]. In a previous report, we demonstrated that the defined sampling density provides a metric which correlates with the error of a linear MLIP, i.e., points with larger sampling densities are, on average, predicted more accurately [42]. We note that that the relationship between model error and sampling density induced by the training is also dependent on the quality/information content of the representation itself. For example, a representation that projects all training configurations into a single point would induce not only a large sampling density but also a uninformed model. As such, the value of the sampling density can not be compared directly from one MLIP to the other but only for each single one of them. In the following, we will therefore only consider the differences and similarities between the sampling density in the training and in the nanoparticle datasets.

Fig. 8.d shows the sampling density induced by the training set on each point in the training set and in the nanoparticle data set, for each of the three tested MLIPs. The sampling density distribution on training and nanoparticle points are essentially equivalent in the case of the PLIP and LBP models which is consistent with the similar error obtained by these linear models for training and nanoparticle configurations. A fairly strong mismatch between sampling densities induced on training or nanoparticle points is instead observed when considering the last layer of the NNP model. In this case, the sampling density induced on training points is, on average, larger than the one induced on nanoparticle points. We conclude that the shift in sampling density distribution parallels the lesser accuracy of the NNP model, when transferred to nanoscale configurations.

V. CONCLUSION

This work provided a quantitative account of transferability which is an important problem associated with the use of MLIP. In this regard, we have analyzed the transferability of three MLIP approaches, namely the LBP, NNP and PLIP models. The MLIPs were trained over relatively small but diverse configurations of ZnO systems. To estimate robust statistics, five different models per MLIP scheme were trained and the deviations among each of these models were computed.

Firstly, the showed that NNP is a better learner than PLIP and LBP when considering the RMSE on both training and test sets configurations. Next, we measured the physical properties including lattice constants, surface energies, and phonon density of states. We showed that both the NNP and PLIP are largely superior than LBP and provide a good description with NNP being slightly superior at this stage than PLIP. Then, by studying molecular dynamics simulation of the liquids, we showed that NNP can lead to unphysical behavior at

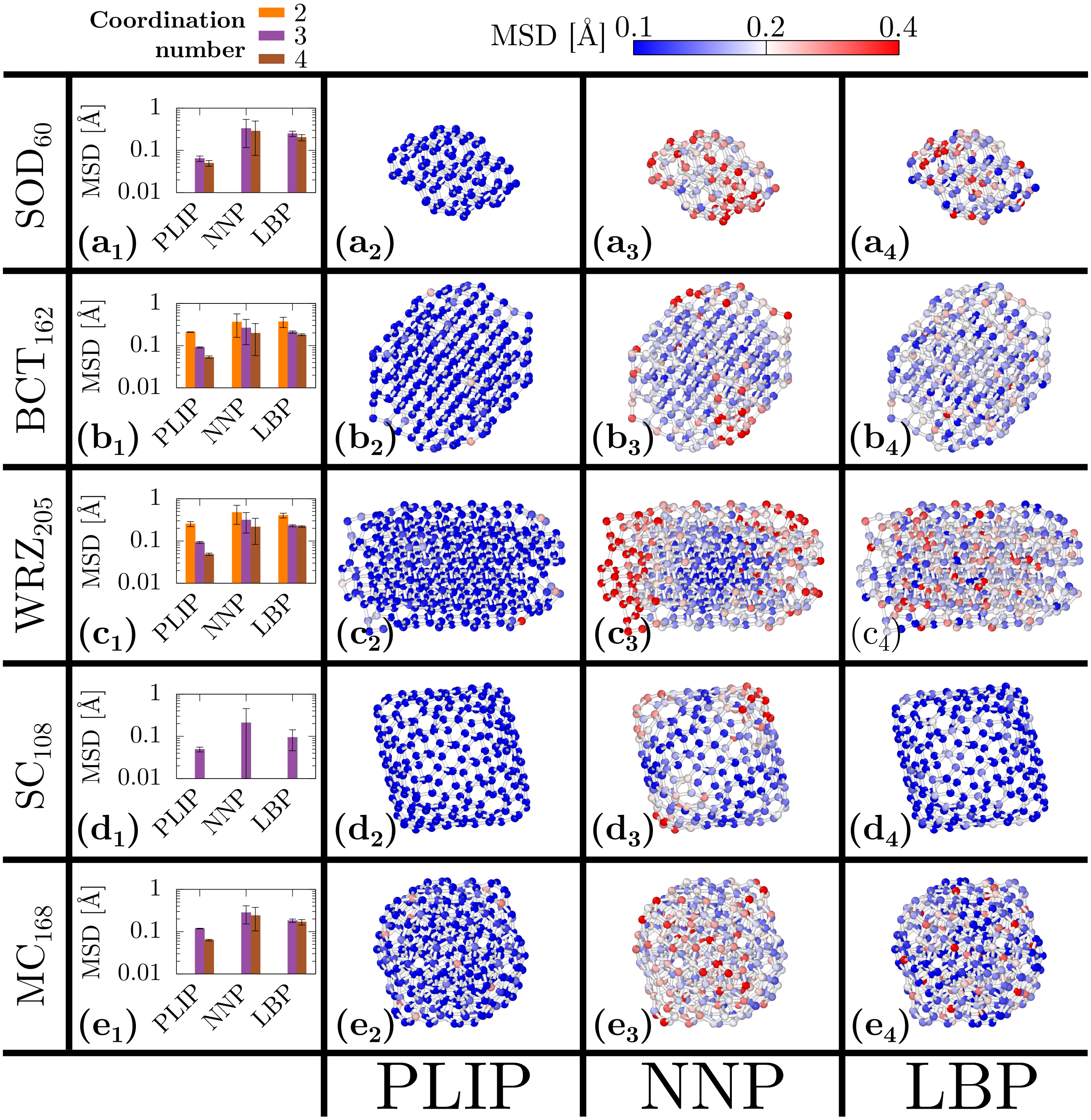
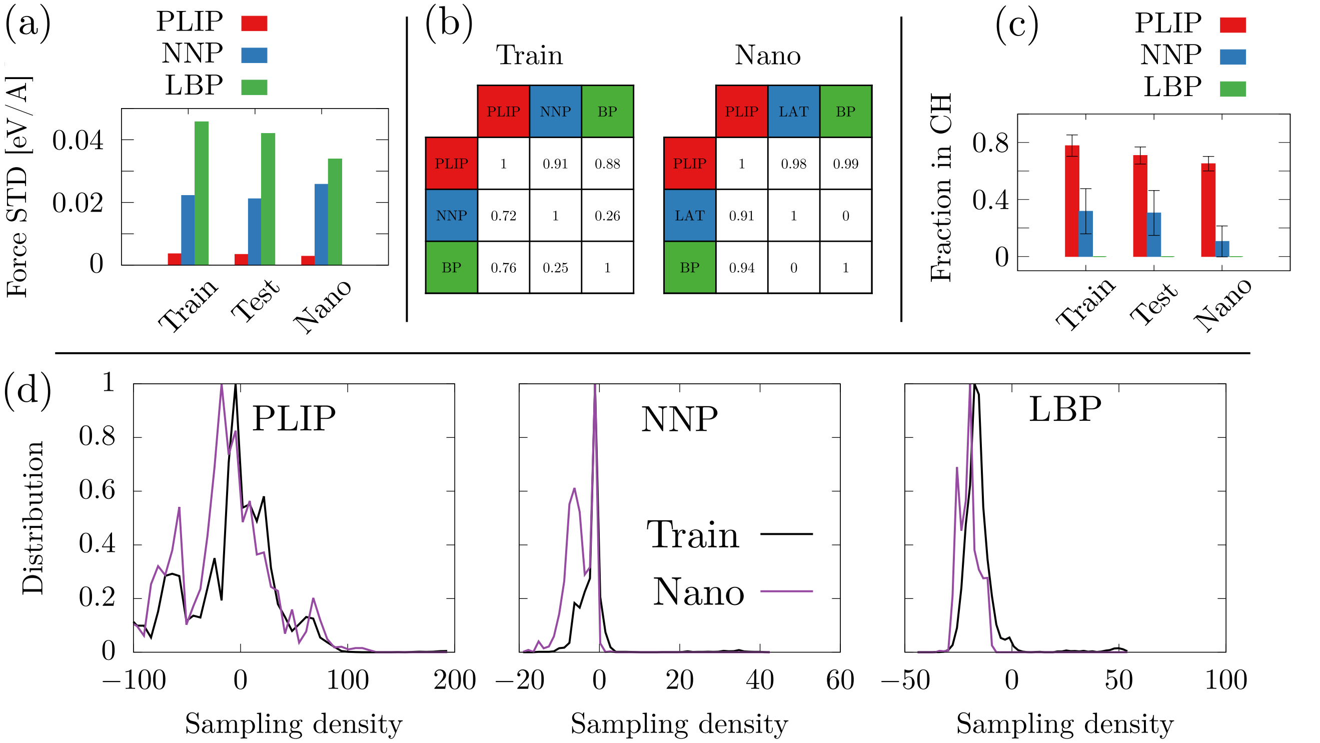


FIG. 7. Influence of the coordination number on the MSD error for five typical nanoparticles and for each considered models. Coordination numbers are computed with a cutoff equal to 2.5 Å.

short distances. In practice, such issues might be fixed by augmenting the database with configuration displaying short interatomic distances or by introducing a prior baseline which is aware of interatomic repulsion at short distances.[59, 60] Finally, we designed a strong transferability test with the structural optimization of nanoparticles. Both linear models outperformed the NNP model. In these example, despite NNP being better learners on the given training-set, they were limited in applicability to systems beyond the training-set. In addition, having LBP being more transferable than NNP despite using the same set of descriptors suggests that the non-linearity brought by the neural network architecture can cause the decrease in transferability. Interestingly, we note that more advanced equivariant message-passing neural network architectures appear to escape from this complexity-transferability trade-offs.[61] Altogether, PLIP by using both a larger set of descriptors and a linear model provides the most accurate and reliable potential in this particular case of ZnO.



Train

	PLIP	NNP	BP
PLIP	1	0.91	0.88
NNP	0.72	1	0.26
BP	0.76	0.25	1

Nano

	PLIP	LAT	BP
PLIP	1	0.98	0.99
LAT	0.91	1	0
BP	0.94	0	1

FIG. 8. (a) Spread in the force predictions obtained using 5 different models. (b) Information imbalance analysis for training and nanoparticle datasets (c) Fraction of points within the training convex hull. (d) Distribution of the sampling density for training and nanoparticle datasets.

In the discussion, we provide further explanations for this observation. Main results of this statistical analysis are that (1) The spread in force prediction shows that NNP provides less consistent results than PLIP especially when transfered into nanoscale configurations, (2) At the level of the information balance, PLIP and NNP are equivalent while being superior than LBP, (3) PLIP's convex hull is the only one providing sufficient overlapping between train, test and nanoparticle points and (4) The nanoparticle data set is much less dense than the the training data set in the NNP models thus suggesting a lack of transferability.

We hope the current work would serve as a template to better understand the possibilities and limitations of different classes of MLIPs, enabling the development of efficient, reliable, and transferable potentials for technologically relevant materials.

VI. ACKNOWLEDGEMENTS

We thank Jacek Goniakowski for fruitful discussions. Computational resources have been provided by the Regional Computer Center CALMIP in Toulouse. This project has been funded in part by the ANR under the grant agreement ANR-21-CE09-0006.

[1] J. Behler, J. Chem. Phys. **145**, 170901 (2016).
[2] V. L. Deringer, M. A. Caro, and G. Csányi, Adv. Mater. **31**, 1902765 (2019).
[3] J. Behler and M. Parrinello, Phys. Rev. Lett. **98**, 146401 (2007).
[4] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Phys. Rev. Lett. **104**, 136403 (2010).
[5] A. Seko, A. Takahashi, and I. Tanaka, Phys. Rev. B **92**, 054113 (2015).
[6] A. Seko, A. Takahashi, and I. Tanaka, Phys. Rev. B **90**, 024101 (2014).
[7] A. Takahashi, A. Seko, and I. Tanaka, Phys. Rev. Materials **1**, 063801 (2017).
[8] A. Seko, A. Togo, and I. Tanaka, Phys. Rev. B **99**, 214108 (2019).
[9] A. M. Goryaeva, J.-B. Maillet, and M.-C. Marinica, Comput. Mater. Sci. **166**, 200 (2019).
[10] M. Benoit, J. Amodeo, S. Combettes, I. Khaled, A. Roux, and J. Lam, Mach. Learn.: Sci. Technol. **2**, 025003 (2020).
[11] A. P. Thompson, L. P. Swiler, C. R. Trott, S. M. Foiles, and G. J. Tucker, J. Comput. Phys. **285**, 316 (2015).
[12] M. A. Wood and A. P. Thompson, J. Chem. Phys. **148**, 241721 (2018).
[13] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K.-R. Müller, Sci. Adv. **3**, e1603015

(2017).

[14] S. Chmiela, H. E. Sauceda, K.-R. Müller, and A. Tkatchenko, Nat. Commun. **9**, 1 (2018).

[15] A. V. Shapeev, Multiscale Model. Simul. (2016).

[16] I. S. Novikov, K. Gubaev, E. V. Podryabinkin, and A. V. Shapeev, Mach. Learn.: Sci. Technol. **2**, 025002 (2020).

[17] R. Drautz, Phys. Rev. B **99**, 014104 (2019).

[18] C. Zeni, K. Rossi, A. Glielmo, and S. de Gironcoli, J. Chem. Phys. **154**, 224112 (2021).

[19] A. Bochkarev, Y. Lysogorskiy, S. Menon, M. Qamar, M. Mrovec, and R. Drautz, Phys. Rev. Mater. **6**, 013804 (2022).

[20] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, and B. Kozinsky, Nat. Commun. **13**, 1 (2022).

[21] I. I. Novoselov, A. V. Yanilkin, A. V. Shapeev, and E. V. Podryabinkin, Comput. Mater. Sci. **164**, 46 (2019).

[22] A. Takahashi, A. Seko, and I. Tanaka, J. Chem. Phys. **148**, 234106 (2018).

[23] C. Zeni, K. Rossi, A. Glielmo, Á. Fekete, N. Gaston, F. Baletto, and A. De Vita, J. Chem. Phys. **148**, 241739 (2018).

[24] V. Botu, R. Batra, J. Chapman, and R. Ramprasad, J. Phys. Chem. C **121**, 511 (2017).

[25] T. Bereau, R. A. DiStasio, A. Tkatchenko, and O. A. von Lilienfeld, J. Chem. Phys. **DETC2018**, 241706 (2018).

[26] H. E. Sauceda, S. Chmiela, I. Poltavsky, K.-R. Müller, and A. Tkatchenko, J. Chem. Phys. **150**, 114102 (2019).

[27] A. P. Bartók, S. De, C. Poelking, N. Bernstein, J. R. Kermode, G. Csányi, and M. Ceriotti, Sci. Adv. **3**, e1701816 (2017).

[28] M. Veit, S. K. Jain, S. Bonakala, I. Rudra, D. Hohl, and G. Csányi, J. Chem. Theory Comput. **15**, 2574 (2019).

[29] T. T. Nguyen, E. Székely, G. Imbalzano, J. Behler, G. Csányi, M. Ceriotti, A. W. Götz, and F. Paesani, J. Chem. Phys. **DETC2018**, 241725 (2018).

[30] A. P. Bartók, M. J. Gillan, F. R. Manby, and G. Csányi, Phys. Rev. B **88**, 054104 (2013).

[31] T. Morawietz, V. Sharma, and J. Behler, J. Chem. Phys. **136**, 064103 (2012).

[32] S. K. Natarajan and J. Behler, Phys. Chem. Chem. Phys. **18**, 28704 (2016).

[33] T. Morawietz and J. Behler, J. Phys. Chem. A **117**, 7356 (2013).

[34] V. L. Deringer and G. Csányi, Phys. Rev. B **95**, 094203 (2017).

[35] A. P. Bartók, J. Kermode, N. Bernstein, and G. Csányi, Phys. Rev. X **8**, 041048 (2018).

[36] M. A. Caro, V. L. Deringer, J. Koskinen, T. Laurila, and G. Csányi, Phys. Rev. Lett. **120**, 166101 (2018).

[37] V. L. Deringer, N. Bernstein, A. P. Bartók, M. J. Cliffe, R. N. Kerber, L. E. Marbella, C. P. Grey, S. R. Elliott, and G. Csányi, J. Phys. Chem. Lett. **9**, 2879 (2018).

[38] V. L. Deringer, M. A. Caro, R. Jana, A. Aarva, S. R. Elliott, T. Laurila, G. Csányi, and L. Pastewka, Chem. Mater. **30**, 7438 (2018).

[39] G. C. Sosso, V. L. Deringer, S. R. Elliott, and G. Csányi, Mol. Simul. **44**, 866 (2018).

[40] J. Lahnsteiner, R. Jinnouchi, and M. Bokdam, Phys. Rev. B **100**, 094106 (2019).

[41] R. Jinnouchi, J. Lahnsteiner, F. Karsai, G. Kresse, and M. Bokdam, Phys. Rev. Lett. **122**, 225701 (2019).

[42] C. Zeni, A. Anelli, A. Glielmo, and K. Rossi, Phys. Rev. B **105**, 165141 (2022).

[43] J. Goniakowski, S. Menon, G. Laurens, and J. Lam, J. Phys. Chem. C **126**, 17456 (2022).

[44] D. J. Binks and R. W. Grimes, Journal of the American Ceramic Society **76**, 2370 (1993).

[45] J. P. Perdew, K. Burke, and M. Ernzerhof, Physical review letters **77**, 3865 (1996).

[46] G. Kresse and D. Joubert, Phys. Rev. B **59**, 1758 (1999).

[47] G. Kresse and J. Hafner, Phys. Rev. B **47**, 558 (1993).

[48] G. Kresse and J. Furthmuller, Phys. Rev. B **54**, 11169 (1996).

[49] A. Singraber, J. Behler, and C. Dellago, J. Chem. Theory Comput. **15**, 1827 (2019).

[50] A. Singraber, T. Morawietz, J. Behler, and C. Dellago, J. Chem. Theory Comput. **15**, 3075 (2019), pMID: 30995035.

[51] S. Plimpton, J. Comput. Phys. **117**, 1 (1995).

[52] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. In 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton, Comput. Phys. Commun. **271**, 108171 (2022).

[53] A. Togo and I. Tanaka, Scr. Mater. **108**, 1 (2015).

[54] F. Viñes, O. Lamiel-Garcia, F. Illas, and S. T. Bromley, Nanoscale **9**, 10067 (2017).

[55] A. Glielmo, C. Zeni, B. Cheng, G. Csányi, and A. Laio, PNAS Nexus **1**, pgac039 (2022).

[56] M. Carli and A. Laio, Mol. Phys. **119**, e1899323 (2021).

[57] A. Rodriguez, M. D'Errico, E. Facco, and A. Laio, J. Chem. Theory Comput. **14**, 1206 (2018).

[58] E. Facco, M. D'Errico, A. Rodriguez, and A. Laio, Sci. Rep. **7**, 1 (2017).

[59] K. Rossi, V. Jurásková, R. Wischert, L. Garel, C. Corminbœuf, and M. Ceriotti, J. Chem. Theory Comput. **16**, 5139 (2020).

[60] S. Thaler and J. Zavadlav, Nat. Commun. **12**, 1 (2021).

[61] C. J. Owen, S. B. Torrisi, Y. Xie, S. Batzner, J. Coulter, A. Musaelian, L. Sun, and B. Kozinsky, arXiv (2023), 10.48550/arXiv.2302.12993, 2302.12993.