Size and temperature transferability of direct and local deep neural networks for atomic forces

Natalia Kuritz,[1] Goren Gordon,[2] and Amir Natan[1,3,*]
[1]*Department of Physical Electronics, Tel Aviv University, Tel Aviv 69978, Israel*
[2]*Department of Industrial Engineering, Tel Aviv University, Tel Aviv 69978, Israel*
[3]*The Sackler Center for Computational Molecular and Materials Science, Tel Aviv University, Tel Aviv 69978, Israel*

(Received 24 April 2018; revised manuscript received 15 August 2018; published 20 September 2018)

A direct and local deep learning (DL) model for atomic forces is presented. We demonstrate the model performance in bulk aluminum, sodium, and silicon and show that its errors are comparable to those found in state-of-the-art machine learning and DL models. We then analyze the model's performance as a function of the number of neighbors included and show that one can ascertain physical attributes of the system from the analysis of the deep learning model's behavior. Finally, we test the size scaling performance of the model and the transferability between different temperatures and show that our model performs well in both scaling to larger systems and high- to low-temperature predictability.

DOI: 10.1103/PhysRevB.98.094109

I. INTRODUCTION

The computation of large systems' atomistic dynamics is required in fields such as biochemistry, surface science, electrochemistry, and many others. *Ab initio* molecular dynamics (AIMD) [1] is a powerful tool but can have a high computational cost which prohibits the computation of large systems for long enough time intervals. A successful approach, with a significantly lower computational cost, is the use of classical force fields (FFs) to model the forces between the atoms [2,3]. This scheme enables the simulation of the dynamics of large systems (more than 10^6 atoms) within the nanosecond and microsecond timescales. The disadvantage of classical models is that they often need system-specific parametrization and cannot handle chemical reactions where molecules break or form new bonds. Another example that can be challenging for a classical FF approach is that of metal oxidation; a metal atom is neutral inside the metal bulk but is charged inside the oxide layer. A possible way to model such varying environments is to introduce more complicated FFs, such as the variable-charge force field [4], charge-optimized many body (COMB) potentials [5] and reactive force-fields (ReaxFF) [6,7]. Such FFs can successfully describe more challenging situations but need greater parametrization and, again, cannot cover all possible atomic configurations.

An approach that was developed in the last decade is to use machine learning (ML) and deep learning (DL) [8] algorithms to build "on-the-fly" computationally cheap predictors for the energy, forces, and other physical properties. This approach enables the performance of calculations with an accuracy that is close enough to fully quantum molecular dynamics (MD) but with running speeds that are more than 100 times faster.

One way to tackle the statistical learning of chemical properties is kernel-based ML. Within this approach the atomic system is represented by physical fingerprints such as the "Coulomb" matrix [9–11], the bag of bonds [12], the bispectrum [13], the smooth overlap of atomic positions [7], bonding angular machine learning [14], tensor representation [15], and more [16–22].

Additional approaches include scattering transforms [23] with the use of wavelets, neural network regression with generalized symmetry functions [24], and feature matrix representation and the idea of covariant kernels, represented with Gaussian processes regression [25,26].

Several DL implementations have been developed recently [27–30]. It is possible to define two main DL algorithm strategies: *convolutional neural network* (CNN) [18,28,31] and *fully connected deep neural network* (DNN) [32–35]. The DNN approach is size extensive and is the simplest option to model energies, as shown in several recent studies.

A desired goal for both the ML and DL approaches is to be able to train the model on a small system and then to use the accrued knowledge in much larger and diverse environments. This requires a formulation of a "local-environment" input to the model.

DL models with a local-environment input were recently suggested by Han *et al.* [36], Zhang *et al.* [30], and Lubbers *et al.* [37]. In these models, the input of the system is presented as a simple function of the atomic positions of each atom's neighbors. The output of the model was the energy, while the forces were estimated from the energy derivatives with respect to the atoms' locations.

In this work, we describe the construction and use of local environments for a DNN-based model for the forces in solids. This model is very close in spirit to the one reported by Han *et al.* [36] but makes a direct prediction of the forces instead of the energy. We first show that with this DNN model we can reach an accuracy that is comparable to that found in state-of-the-art ML and DL models. We demonstrate this for bulk Al, Si, and Na at temperatures of 300 and 2000 K. We then analyze the dependence of the error on the number of neighbors used for the input and show that physical attributes of the underlying system can be learned from this analysis.

*amirnatan@post.tau.ac.il

TABLE I. Force component MAE comparison for all systems. In all of the cases in this table the temperature and the cell size are the same in the training and validation sets. The values in the literature differ in temperatures and the definitions of error criteria. Reference [19] does not specify temperature and reports the forces' MAE. Reference [61] uses $T = 300$ K and $T = 600$ K and reports the root-mean-square error (RMSE) for the force components and not the MAE (RMSE is typically a bit higher than the MAE for the same data). Reference [25] uses $T = 300$ K and $T = 800$ K; Ref. [26] uses $T = 1000$ K and calculates the error for the whole force vector (should be divided by 2 to compare to the component MAE). A more detailed analysis is shown in the SM [60].

Atom	Training temperature (K)	Unit cell size	K points	MAE, this work (ev/Å)	Literature values (ev/Å)
Al	2000	27	125	0.025	0.02 [19]
	300	27	125	0.022	0.02 [61]
Na	2000	27	125	0.014	
	300	27	125	0.004	
	2000	16	8	0.058	0.08 [61]
Si	300	16	8	0.030	0.1 [25,26]

Finally, we analyze the ability of the developed DL model to do actual size scaling, that is, to use training in a small cell for prediction in larger cells. We also test the ability to operate at different temperatures, i.e., to train at one temperature and predict at another temperature. We conclude with a discussion of how to proceed and build fully scalable and transferable DL models that can work with a wide range of environments. The rest of the paper is organized as follows: we first describe the model, methods, and data sets; we then show the results for systems with the same size, different sizes, and different temperature. Finally, we discuss the meaning of some of the observations and the challenges faced in developing a fully scalable prediction.

II. METHODS

A. Tools

We used the VASP [38–40] package for all quantum simulations of bulk Al and bulk Na. We analyzed VASP files to get the radial distribution function (RDF) using the AFLOW [41] package. We used the following PYTHON libraries for building the network structure and for the training: NUMPY and SCIPY [42], PANDAS [43], IPYTHON [44], MATPLOTLIB [45], TENSORFLOW [46], TFLEARN [47], ASE [48] and SCIKIT-LEARN [49]. All the code was written in PYTHON. In addition, the MATLAB software [50] was used to draw most of the figures.

B. Data sets

We used the following bulk supercells for the training and validation of Al and Na: a $3 \times 3 \times 3$ supercell (27 atoms) and $5 \times 5 \times 5$ supercell (125 atoms). For Si, we used supercells of $2 \times 2 \times 2$ (16 atoms) and $4 \times 4 \times 4$ (128 atoms).

The training set consisted of about 1620 MD steps that we randomly chose from a trajectory of 1800 steps. The other 180 configurations were used for validation. When changing the supercell size $M \times M \times M$, we changed the number of k points, defined by a Monkhorst-Pack [51] grid $M_k \times M_k \times M_k$, so that the Born–von Kármán cell [52] stays roughly constant (i.e., $M \times M_k$ is kept constant) and so the level of electronic sampling is similar.

The data sets were prepared with the following protocol. First, the supercells were built for each material: fcc for Al, bcc for Na, and diamond structure for Si with the experimental lattice constants 4.05, 4.29, and 5.43 Å, respectively [53]. Then an AIMD was run with VASP for Na and Si, and classical MD was run for Al with the EMT force field and the ASE package. The MD was performed with a constant supercell volume and shape to produce the relevant atomic positions. For Al, the atomic configuration was recorded each 50 fs. For Na and Si, the atomic configuration was recorded each 1 fs. The MD time propagation was performed in the canonical ensemble and with the Nosè algorithm for Si and Na. The Al run was conducted in the microcanonical ensemble with temperatures around $T = 300$ K and $T = 2000$ K. For the final calculation of the forces, we applied the following protocol. Each of the produced structures was run with density functional theory (DFT) [54] without further geometrical relaxation; we used the Perdew-Burke-Ernzerhof functional [55] without spin polarization, the VASP projector augmented-wave pseudopotentials [56,57], and an energy cutoff of 260 eV for Si and Na and 520 eV for Al, which was found to be sufficient for the accuracies we report later. The number of k points for each of the cells is shown in Table I.

C. Deep learning models and learning procedure

In this section, we describe the structure of the DL model, as well as its input and output.

Network architecture. For each atom, the output of the model is a three-dimensional vector of the Cartesian forces. For each atom, we find the N nearest neighbors (we used $N = 13$ in most simulations); the convergence of errors with respect to N is discussed later in the text. We sort the atoms according to their distance (closest is first) and then assign for each neighbor the following quantities: $(\mathbf{d}_n, 1/d_n, 1/d_n^2)$, where d_n is the scalar distance. In this work we analyze only monoatomic systems, so the atomic number Z_n is irrelevant and does not contribute to the model.

The input is fed to a fully connected neural network with two hidden layers. The first hidden layer has L nodes, and the second hidden layer has $L/3$ nodes. L was typically around 3800 for Si and around 250 for Al and Na; we analyze the model mean absolute error (MAE) sensitivity as a function of

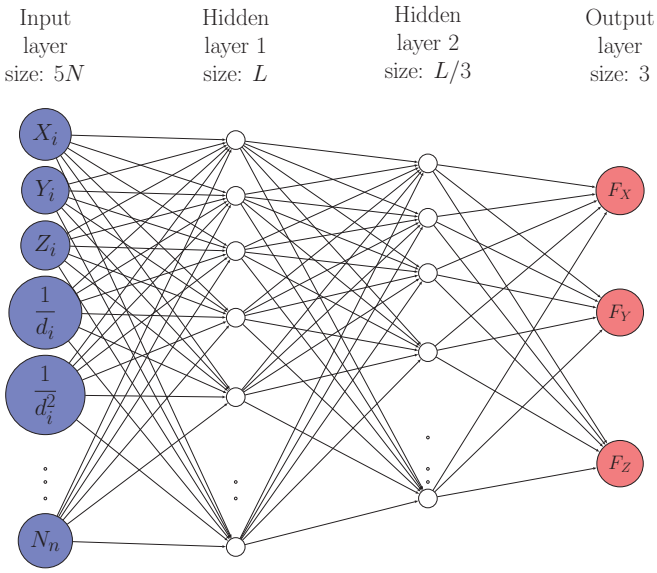

FIG. 1. Schematic illustration of the NN model. The input layer consists of the distance vectors of the atom from its nearest neighbors. The output layer has three nodes with the values of the forces in each direction.

L later. Finally, the output corresponds to the three Cartesian forces. We used the CRELU[58] function for activation of the two first layers and a linear function for the output layer. We tried more hidden layers and larger hidden layers but found that this did not improve the model accuracy much. A typical model architecture is illustrated in Fig. 1.

Network training and optimization. We used the adaptive moment estimation (Adam) [59] algorithm to train the model against the results of quantum calculations for the forces. We used a learning rate of 0.0001, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and minibatches of 100 samples.

Rotational symmetry. The model, as defined above, is not rotationally invariant; hence, it will have significantly higher errors when the data are rotated. To solve this we have added rotation operations, applied during the training, which bring the first neighbor to the $\hat{x}$ axis and the second neighbor to the xy plane. Those rotations are then applied for the input of the test data, and the inverse rotation is applied for the model prediction. With that, the model becomes rotationally invariant. We describe the transformation in the Supplemental Material (SM) [60] and also analyze the effect of this transformation on the model performance and show that it does not lead to larger errors.

III. RESULTS

A. Model performance for aluminum, sodium, and silicon

In this section, we show the model performance when the cell and temperature are the same for the training and validation of the model. We used cells of 27 atoms for Al and Na and 16 atoms for Si. The cells were trained and tested at 300 and 2000 K. The results are shown in Fig. 2 and demonstrate that a small enough MAE was achieved. The MAEs are also listed in Table I. We also add in the SM [60] an analysis of the force vector error, which was found to be around two times the force components' MAE.

A comparison of these results with those from previous studies shows that the presented DL models reach sufficient

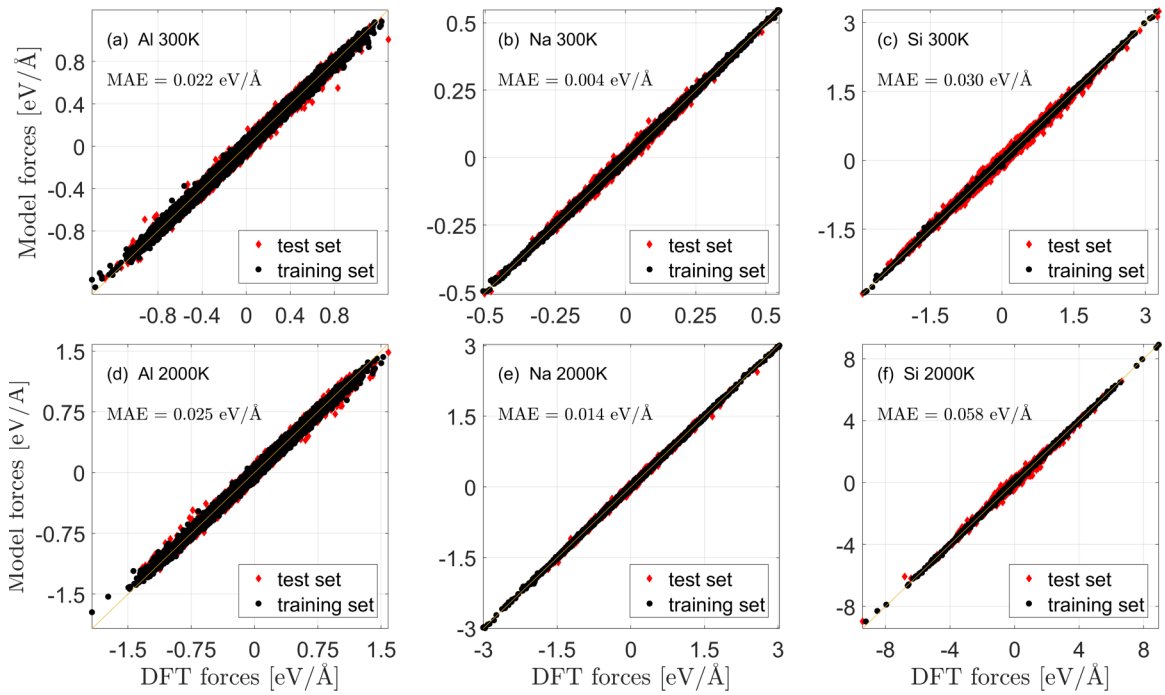

FIG. 2. Comparison of the estimated forces to DFT forces at (a)–(c) 300 K and (d)–(f) 2000 K. Results are shown for (a) and (d) Na (27 atoms), (b) and (e) Al (27 atoms), and (c) and (f) Si (16 atoms). The training set results are shown with black dots, while the test set results are shown with red diamonds. The solid line is a result of linear fit between the model and DFT results.

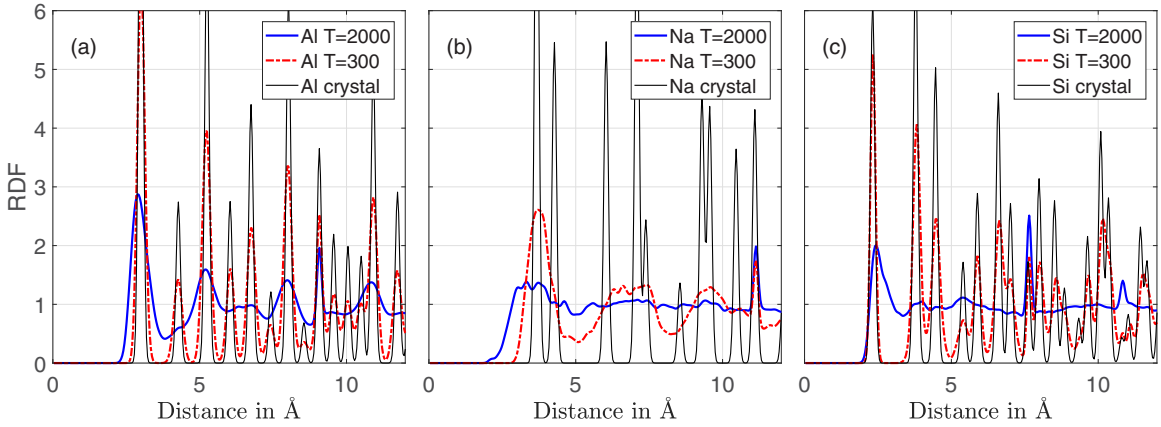

FIG. 3. Average radial distribution function (RDF) of the different structures: (a) Al, (b) Na, and (c) Si. The RDF of the ground-state crystal is shown with a solid black line, the RDF at $T = 300$ K is shown with a dash-dotted red line, and the $T = 2000$ RDF is shown with a solid blue line.

accuracy for the atomic forces. Such an accuracy was shown to allow running MD simulations without any precalculated FFs at an accuracy that is close to that of AIMD [36,62]. It should be noted that the inclusion of the $1/d_n$ and $1/d_n^2$ terms was important to achieve reasonable MAEs. We show in the SM [60] that we can reach similar results by adding either $1/d_n$, d_n, or the combination of $1/d_n$ and $1/d_n^2$ and that supplying only the distance vector $\mathbf{d_n}$ results in significantly higher errors. Our interpretation of this is that the distance norm is an important parameter that "helps" the model to locate the radial position on the energy surface. Supplying either d_n or $1/d_n$ results in a similar performance. The inclusion of additional terms such as $1/d_n^6$ and $1/d_n^{12}$ did not help to further improve the results.

B. Sensitivity to the number of neighbors

In this section we analyze the MAE dependence on the number of neighbor atoms that were used in the model. We performed this analysis both at 300 and 2000 K. At 300 K all three materials are solids. The melting points for Si, Al, and Na are 1687, 933, and 371 K, respectively [63]. Since we conducted the simulations at a constant volume, this is an underestimation, and we can assume that the materials are somewhere between solid and liquid. A possible measure of the material atomic structure is the RDF. Figure 3 shows the normalized RDF of the different systems at both 300 and 2000 K. At 300 K, both Si and Al exhibit an RDF that is close to the crystalline system. In contrast, although for Na it still shows peaks that are related to the crystalline structure, it is already heavily smeared. At 2000 K, all systems are heavily smeared, with Al still showing some structural peaks. It is very clear that at 2000 K there are many distances that do not appear at 300 K.

We expected that systems with more nearest neighbors (NNs), like Al (fcc, 12 NNs), would generally require more neighbors to converge in comparison to systems like Na (bcc, 8 NNs) and Si (diamond, 4 NNs). Furthermore, we expected that this trend would be clearer at the lower temperature, where all the materials are solids. Figure 4(a) shows the results for 300 K, and Fig. 4(b) gives the results for 2000 K. It is evident that at both temperatures Si and Na converge faster than Al. Si converges slightly faster than Na at 2000 K but reaches a significantly higher converged MAE. The trends are, in fact, clearer at the higher temperature; one possible reason is that the AIMD at that temperature covered a wider range of configurations. Further analysis of this trend will follow in future work.

This analysis, namely, of MAE as a function of NNs, makes it possible to uncover physical attributes of the system in question from the DL algorithm. While in the systems we analyzed the physical attributes are known, we suggest that for more complex systems, such an analysis can give new insight into the internal structure of the system.

C. Temperature analysis

Here, we analyzed the ability of a model that was trained at one temperature to predict results at another temperature. Naturally, if we use low-temperature MD as our training

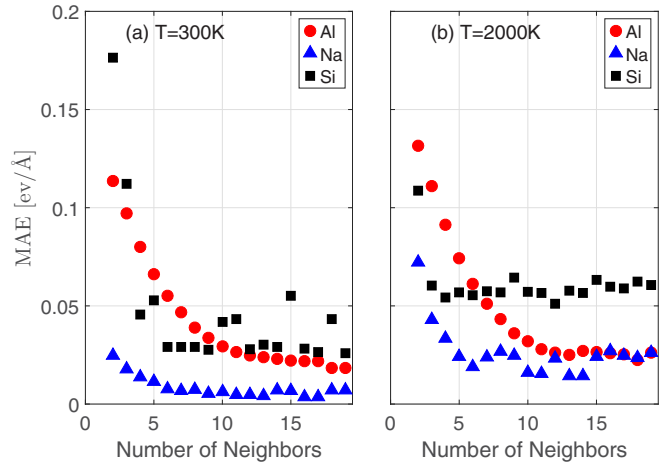

FIG. 4. Mean absolute error (MAE) as a function of the number of input neighbor atoms that are used for the model [data produced (a) at 300 K and (b) at 2000 K]. Al with red circles, Na is shown with blue triangles, and Si is shown with black squares.

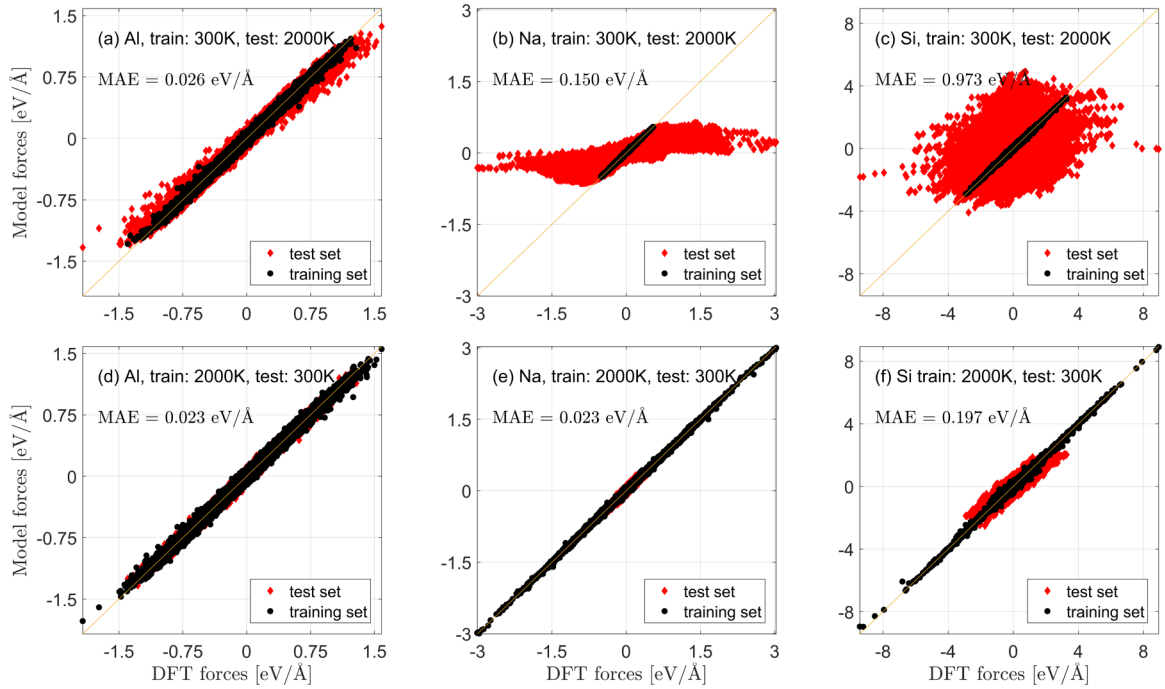

FIG. 5. Comparison of the estimated forces to DFT forces at (a)–(c) 300 K and (d)–(f) 2000 K. Results are shown for (a) and (d) Na (27 atoms), (b) and (e) Al (27 atoms), and (c) and (f) Si (16 atoms). The training set results are shown with black dots, while the test set results are shown with red diamonds. The solid line is a result of linear fit between the model and DFT results.

set, there is a high chance that most of the set will be around the ground-state minimum energy. Therefore, the outcome might not predict well other metastable minima of the potential-energy surface. Furthermore, as is obvious from Fig. 3, the $T = 300$ K simulation does not always deviate enough from the ground-state crystal, so some distances that exist at $T = 2000$ K are completely absent at $T = 300$ K. If we use high-temperature MD results as our training set, we have a higher chance to cover more configurations, but we might have sparser coverage for each minimum. Consequently, it is natural to assume that training at higher temperatures might lead to performing well at lower temperatures, while the opposite is less probable. In Fig. 5 we show the results of all systems for $300 \rightarrow 2000$ K and for $2000 \rightarrow 300$ K. It is very clear that the first case, training at low and testing at high temperature, yields poor performance; this is especially true for Si, where it is clear that the model seems to be almost random. The second case, training at high temperature and testing at low temperature, gives a higher MAE in comparison to the same temperature tests but behaves reasonably well and can be used. This result shows that it is possible to construct temperature-transferable DL models for MD simulations.

D. Scaling analysis

In this section we evaluate the ability of a model that was trained with a cell of a given size to perform with cells that are larger. As we use a local environment for the training input, the model is, in a way, "blind" to the number of the atoms in the cell. Clearly, there can be long-range forces that a local environment will not capture. A similar problem can also exist in classical FFs that do not include polarization terms and have a cutoff. A full solution to the problem of scaling will require some specific treatment or training for the response to long-range forces and is beyond the scope of this work. We can still hope that the local model can yield reasonably good force predictions in many scenarios. In Fig. 6 we show the prediction of forces for the three materials. For Al (2000 K), training with 27 atoms and testing with 125 atoms yields an MAE of $\sim$0.03 eV/Å, which is comparable with the same-size performance. For Na (2000 K), training with 27 atoms and testing with 125 atoms yields an MAE of $\sim$0.03 eV/Å, which is a bit worse than the same-size model performance. For Si (300 K), going from 16 atoms to 128 atoms yields an MAE of 0.12 eV/Å, higher than the same-size performance of 0.03 eV/Å.

It is evident that reasonable scaling was demonstrated for Al and Na, which means that we could use the smaller cell to estimate errors in the larger cell. With Si, further work should be done, as the performance penalty is a bit too high.

IV. NETWORK ARCHITECTURE ANALYSIS

To study the required network size, we checked the MAE's sensitivity to the number of hidden layers and the size of the first layer. In the first test, we used two hidden layers, the first with L nodes and the second with $L/3$ nodes, and we varied L. This parameter can strongly affect the model computational efficiency. Since we have two layers, we can expect the model computation time to have $O(L^2)$ scaling. Figure 7(a) demonstrates that for $T = 300$ K, in Na and Al, we can reduce L to $\sim 250 \simeq 20N$ ($N = 13$ being the number of neighbors that are used) without significantly increasing the MAE. With Si, even at $T = 300$ K, there is an improvement

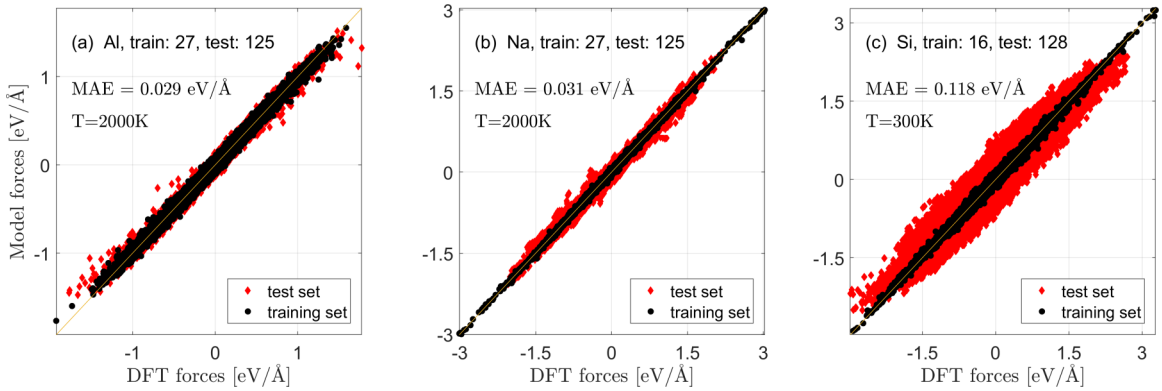

FIG. 6. Comparison of estimated forces and DFT forces when the sizes of the training system and validation system are not the same. The following sets are shown: (a) Al, train with 27 atoms and validation with 125 atoms, (b) Na, train with 26 atoms and test with 125 atoms, and (c) Si, train with 16 atoms and test with 128 atoms. The training set results are shown with black dots, while the test set results are shown with red diamonds. The solid line is a result of linear fit between the model and DFT results.

in performance when increasing L to ∼2000. The picture at $T = 2000$ K, shown in Fig. 7(b), is slightly different. For Si, we see significant but slow improvement in performance when increasing L; however, for L above 2000 the algorithm starts to have convergence problems and does not always find the minimal possible MAE. A larger L means more degrees of freedom for the model and hence theoretically lower MAE; in practice, at some point there are too many degrees of freedom for a given set of data, and hence, convergence becomes more difficult. In Si and Na, it is evident that the improvement with L is more significant at the higher temperature. An essential physical reason for the ability to use small L at low temperatures is the following. As the temperature becomes low, the deviations from equilibrium can be described mostly within the harmonic approximation. Hence, the calculated quantum forces become linear with the distance vector. A description of a linear transformation of the distances requires a minimal L, so it is easy to build a small model for the force prediction. As the temperature becomes high, there is a significant deviation from the harmonic approximation, and hence, the energy function becomes more complicated. This more complicated energy surface requires more domains of piecewise linearity for the forces and therefore a larger L.

In the second test, we used $L \simeq 200N$ and checked whether increasing the number of hidden layers helps to improve the results. In this test, the first hidden layer was with L nodes, and all the next layers had $L/3$ nodes. As is evident from Fig. 8, increasing the number of hidden layers beyond 2 does not improve the error. This trend is true for both $T = 300$ K and $T = 2000$ K. Increasing the number of hidden layers beyond 6 resulted in overfitting problems, probably because more data were needed for the number of parameters that are fitted.

We also checked the model sensitivity for the number of nodes in the second hidden layer L_2, which was initially set to $L/3$. We show this analysis in the SM [60]; we also show that we can build a reasonably small model with only a single hidden layer and with similar performance.

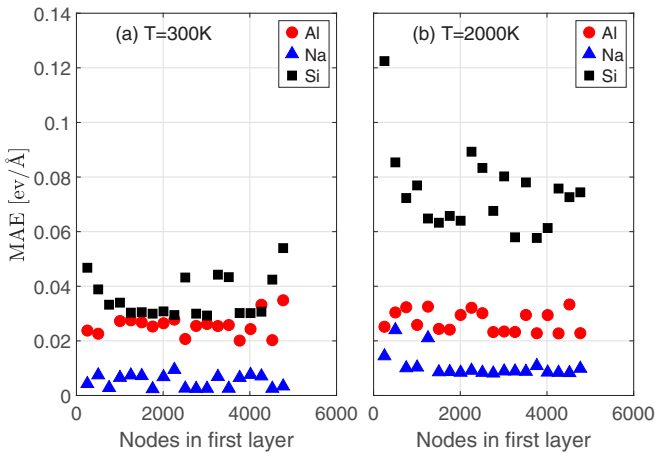

FIG. 7. MAE dependence on the number of nodes L of the first hidden layer, (a) showing $T = 300$ K and (b) showing $T = 2000$ K. Al is shown with red circles, Na is shown with blue triangles, and Si is shown with black squares.

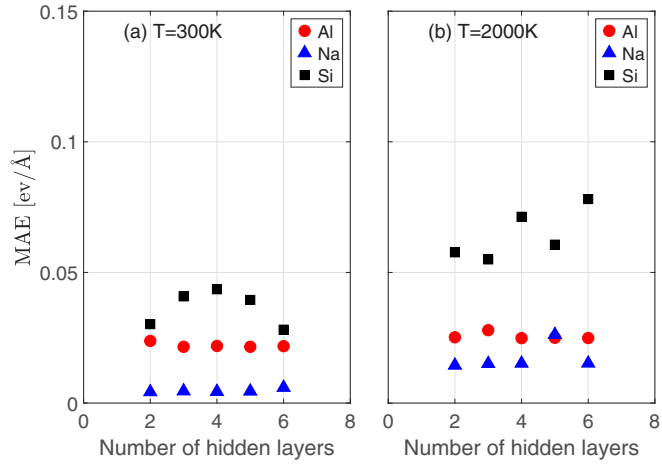

FIG. 8. MAE dependence on the number of hidden layers. The x axis is the number of hidden layers, while the y axis is the model's MAE: (a) $T = 300$ K and (b) 2000 K. Al is shown with red circles, Na is shown with blue triangles, and Si is shown with black squares.

V. SUMMARY AND DISCUSSION

In this work, we presented and implemented a new DL model for atomic forces. This model enables predicting directly atomic forces with a close to DFT accuracy while using a MD sampling learning procedure. We also studied the model properties and the physical insights that this model provides. As the model predicts the forces directly and not as an energy derivative, it has the potential of being faster and more accurate than models that calculate the energy and then use numerical derivatives to find the forces. However, it is possible to argue that since it is not an explicit energy derivative, it might not conserve energy. A possible and simple solution is to predict also the energy and use it in the MD thermostat. An approach that combines direct force prediction and energy prediction was demonstrated with kernel-ridge regression by Botu and Ramprasad [19] and Huan *et al.* [61].

DL and ML models differ from classical FFs by not assuming an explicit physical model for the forces. We can therefore expect that their transferability behavior from one temperature to another might be different as well. In this work, we used MD as the sampling method to construct the training and test sets. We showed that sampling at low temperatures does not have sufficient coverage of the configuration space, so the trained model cannot predict the behavior at a higher temperature. In contrast to low-temperature sampling, once we use a training set constructed from high-temperature MD runs, we can use the model at lower temperatures with reasonable accuracy. This can be explained by a better sampling of the configuration space at higher temperatures. We expect that other DL and ML models will behave similarly, as the lack of any assumption of an explicit physical model for the forces is common to most of them.

Another significant observation is the scalability property of the suggested model. As we showed above, for Al and Na, one can study relatively small systems such as a $3 \times 3 \times 3$ unit cell (27 atoms) to gain knowledge about much larger $5 \times 5 \times 5$ unit cells (125 atoms). We demonstrated this in Fig. 6, which shows results for the transition from small to large cells. For Al and Na, this was shown for data sets that were produced at 2000 K, which can have forces that are significantly beyond the harmonic regime. Temperature and size transferability was also demonstrated for Si with ML and kernel-ridge regression by Suzuki *et al.* [22]. They showed size transferability from 64 atoms to 512 atoms and temperature transferability in the range of 300 to 1650 K.

The analysis of the model sensitivity to the number of neighbors that are used, as shown in Fig. 4 for $T = 300$ K and $T = 2000$ K, demonstrates that we can learn some physical properties of the system (e.g., the atom coordination) from the model performance. Some qualitative trends are evident from the graphs; first, at low temperatures, even the first neighbor can produce a reasonable estimation of the forces in the metals Al and Na. However, Si, which has four covalent bonds, needs at least four neighbors at both checked temperatures to estimate reasonably the atomic forces. At the high temperature, more than 12 neighbors are required for Al, and more than 6 neighbors are needed for Na, where 12 and 8 are the numbers of first-nearest neighbors for the fcc Al and bcc Na structures.

To summarize, the presented model demonstrates two essential properties: size scalability and temperature transferability. Size scalability means that one can predict atomic forces of large systems while learning from small systems that one can study by DFT. Temperature transferability of a force field is extremely important for MD simulations when one would like to find phase transitions and temperature-dependent processes. Obviously, there are also limitations in this method; first of all, a local predictor will probably underperform in situations where long-range forces, not captured by the model, dominate the picture. Furthermore, it is evident that the performance is not equally good for different materials, with Si seen to be more challenging, and this might require a more complicated model. Finally, we have checked a relatively homogeneous environment, and while the environment is local, there might be a need for significant additional training in situations that include interfaces and surfaces.

ACKNOWLEDGMENTS

A.N. acknowledges support from the Pazy Foundation (Grant No. 281/18) and the Planning & Budgeting Committee of the Council of High Education and the Prime Minister's Office of Israel, in the framework of the INREP project. G.G. acknowledges support from the Jacobs Foundation.

[1] D. Marx and J. Hutter, *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods* (Cambridge University Press, Cambridge, 2012).

[2] D. Allen and M. P. Tildesley, *Computer Simulation of Liquids*, Oxford Science Publications (Clarendon, Oxford, 1989).

[3] B. Frenkel and D. Smit, *Understanding Molecular Simulation: From Algorithms to Applications* (Elsevier, London, UK, 2001).

[4] F. G. Sen, A. Kinaci, B. Narayanan, S. K. Gray, M. J. Davis, S. K. R. S. Sankaranarayanan, and M. K. Y. Chan, J. Mater. Chem. A **3**, 18970 (2015).

[5] T. Liang, B. Devine, S. R. Phillpot, and S. B. Sinnott, J. Phys. Chem. A **116**, 7976 (2012).

[6] T. Liang, Y. K. Shin, Y.-T. Cheng, D. E. Yilmaz, K. G. Vishnu, O. Verners, C. Zou, S. R. Phillpot, S. B. Sinnott, and A. C. van Duin, Annu. Rev. Mater. Res. **43**, 109 (2013).

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

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, MA, 2016).

[9] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, Phys. Rev. Lett. **108**, 058301 (2012).

[10] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, New J. Phys. **15**, 095003 (2013).

[11] K. Hansen, G. Montavon, F. Biegler, S. Fazli, M. Rupp, M. Scheffler, O. A. von Lilienfeld, A. Tkatchenko, and K. R. Müller, J. Chem. Theory Comput. **9**, 3404 (2013).

[12] K. Hansen, F. Biegler, R. Ramakrishnan, W. Pronobis, O. A. von Lilienfeld, K. R. Müller, and A. Tkatchenko, J. Phys. Chem. Lett. **6**, 2326 (2015).

[13] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Phys. Rev. Lett. **104**, 136403 (2010).

[14] B. Huang and O. A. von Lilienfeld, J. Chem. Phys. **145**, 161102 (2016).

[15] H. Huo and M. Rupp, arXiv:1704.06439.

[16] O. A. von Lilienfeld, R. Ramakrishnan, M. Rupp, and A. Knoll, Int. J. Quantum Chem. **115**, 1084 (2015).

[17] K. T. Schütt, H. Glawe, F. Brockherde, A. Sanna, K. R. Müller, and E. K. U. Gross, Phys. Rev. B **89**, 205118 (2014).

[18] K. T. Schütt, F. Arbabzadah, S. Chmiela, K. R. Müller, and A. Tkatchenko, Nat. Commun. **8**, 13890 (2017).

[19] V. Botu and R. Ramprasad, Int. J. Quantum Chem. **115**, 1074 (2015).

[20] F. A. Faber, A. Lindmaa, O. A. von Lilienfeld, and R. Armiento, Phys. Rev. Lett. **117**, 135502 (2016).

[21] T. Mueller, A. G. Kusne, and R. Ramprasad, *Reviews in Computational Chemistry*, edited by A. L. Parrill and K. B. Lipkowitz, Vol. 29 (Wiley, New York, 2016) Chap. 4, pp. 186–273.

[22] T. Suzuki, R. Tamura, and T. Miyazaki, Int. J. Quantum Chem. **117**, 33 (2017).

[23] M. Hirn, N. Poilvert, and S. Mallat, arXiv:1502.02077.

[24] J. Behler, J. Chem. Phys. **134**, 074106 (2011).

[25] Z. Li, J. R. Kermode, and A. De Vita, Phys. Rev. Lett. **114**, 096405 (2015).

[26] A. Glielmo, P. Sollich, and A. De Vita, Phys. Rev. B **95**, 214302 (2017).

[27] M. Gastegger, J. Behler, and P. Marquetand, Chem. Sci. **8**, 6924 (2017).

[28] G. B. Goh, C. Siegel, A. Vishnu, N. Hodas, and N. Baker, in *IEEE Winter Conference on Applications of Computer Vision (WACV), 2018* (IEEE, Piscataway, NJ, 2018), pp. 1340–1349.

[29] J. Behler, Angew. Chem., Int. Ed. **56**, 12828 (2017).

[30] L. Zhang, J. Han, H. Wang, R. Car, and Weinan E, Phys. Rev. Lett. **120**, 143001 (2018).

[31] G. B. Goh, N. O. Hodas, and A. Vishnu, J. Comput. Chem. **38**, 1291 (2017).

[32] J. Behler, Int. J. Quantum Chem. **115**, 1032 (2015).

[33] J. Behler, J. Phys.: Condens. Matter **26**, 183001 (2014).

[34] M. Gastegger and P. Marquetand, J. Chem. Theory Comput. **11**, 2187 (2015).

[35] M. Gastegger, C. Kauffmann, J. Behler, and P. Marquetand, J. Chem. Phys. **144**, 194110 (2016).

[36] J. Han, L. Zhang, R. Car, and Weinan E, arXiv:1707.01478.

[37] N. Lubbers, J. S. Smith, and K. Barros, J. Chem. Phys. **148**, 241715 (2018).

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

[39] G. Kresse and J. Hafner, Phys. Rev. B **49**, 14251 (1994).

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

[41] S. Curtarolo, W. Setyawan, G. L. W. Hart, M. Jahnatek, R. V. Chepulskii, R. H. Taylor, S. Wang, J. Xue, K. Yang, O. Levy, M. J. Mehl, H. T. Stokes, D. O. Demchenko, and D. Morgan, Comput. Mater. Sci. **58**, 218 (2012).

[42] S. van der Walt, S. C. Colbert, and G. Varoquaux, Comput. Sci. Eng. **13**, 22 (2011).

[43] W. McKinney, *Proceedings of the 9th Python in Science Conference*, edited by S. van der Walt and J. Millman (SciPy, Austin, Texas, 2010), pp. 51–56.

[44] F. Perez and B. E. Granger, Comput. Sci. Eng. **9**, 21 (2007).

[45] J. D. Hunter, Comput. Sci. Eng. **9**, 90 (2007).

[46] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, TensorFlow: Large-scale machine learning on heterogeneous systems (2015), software available from tensorflow.org.

[47] A. Damien *et al.*, TFLEARN, https://github.com/tflearn/tflearn.

[48] A. Larsen, J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dulak, J. Friis, M. N. Groves, B. B. Hammer, C. Hargus, A. Hjorth Larsen, J. Jørgen Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, M. Dułak, J. Friis, M. N. Groves, B. B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. Bjerre Jensen, J. Kermode, J. R. Kitchin, E. Leonhard Kolsbjerg, J. Kubal, K. Kaasbjerg, S. Lysgaard, J. Bergmann Maronsson, T. Maxson, T. Olsen, L. Pastewka, A. Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, T. Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, and K. W. Jacobsen, J. Phys.: Condens. Matter **29**, 273002 (2017).

[49] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and É. Duchesnay, J. Mach. Learn. Res. **12**, 2825 (2012).

[50] MATLAB, version 9.3.0.713579 (R2017b), The Mathworks, Inc., Natick, MA, 2017.

[51] H. J. Monkhorst and J. D. Pack, Phys. Rev. B **13**, 5188 (1976).

[52] N. W. Ashcroft and N. D. Mermin, *Solid State Physics* (Saunders College, Philadelphia, 1976).

[53] R. Wyckoff, *Crystal Structures* (Wiley, New York, 1963), Vol. 1.

[54] W. Koch and M. Holthausen, *A Chemist's Guide to Density Functional Theory* (Wiley-VCH, Weinheim, 2000).

[55] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996).

[56] P. E. Blöchl, Phys. Rev. B **50**, 17953 (1994).

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

[58] W. Shang, K. Sohn, D. Almeida, and H. Lee, in *Proceedings of the 33rd International Conference on Machine Learning, ICML'16*, Vol. 48 (ACM, New York, NY, USA, 2016), pp. 2217–2225.

[59] D. P. Kingma and J. Ba, arXiv:1412.6980.

[60] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevB.98.094109 for additional details.

[61] T. D. Huan, R. Batra, J. Chapman, S. Krishnan, L. Chen, and R. Ramprasad, npj Comput. Mater. **3**, 37 (2017).

[62] K. Shakouri, J. R. Behler, J. Meyer, and G.-J. Kroes, J. Phys. Chem. Lett **8**, 2131 (2017).

[63] D. Lide, *CRC Handbook of Chemistry and Physics*, 84th ed. (Taylor and Francis, Boca Raton, Florida, 2003).