Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics

Linfeng Zhang and Jiequn Han
Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544, USA

Han Wang*
Institute of Applied Physics and Computational Mathematics, Fenghao East Road 2, Beijing 100094, People's Republic of China and CAEP Software Center for High Performance Numerical Simulation, Huayuan Road 6, Beijing 100088, People's Republic of China

Roberto Car
Department of Chemistry, Department of Physics, Program in Applied and Computational Mathematics, Princeton Institute for the Science and Technology of Materials, Princeton University, Princeton, New Jersey 08544, USA

Weinan E†
Department of Mathematics and Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544, USA and Center for Data Science, Beijing International Center for Mathematical Research, Peking University, Beijing Institute of Big Data Research, Beijing 100871, People's Republic of China

(Received 3 August 2017; published 4 April 2018)

We introduce a scheme for molecular simulations, the deep potential molecular dynamics (DPMD) method, based on a many-body potential and interatomic forces generated by a carefully crafted deep neural network trained with *ab initio* data. The neural network model preserves all the natural symmetries in the problem. It is first-principles based in the sense that there are no *ad hoc* components aside from the network model. We show that the proposed scheme provides an efficient and accurate protocol in a variety of systems, including bulk materials and molecules. In all these cases, DPMD gives results that are essentially indistinguishable from the original data, at a cost that scales linearly with system size.

DOI: 10.1103/PhysRevLett.120.143001

Molecular dynamics (MD) is used in many disciplines, including physics, chemistry, biology, and materials science, but its accuracy depends on the model for the atomic interactions. *Ab initio* molecular dynamics (AIMD) [1,2] has the accuracy of density functional theory (DFT) [3], but its computational cost limits typical applications to hundreds of atoms and time scales of ∼100 ps. Applications requiring larger cells and longer simulations are currently accessible only with empirical force fields (FFs) [4–6], but the accuracy and transferability of these models is often in question.

Developing FFs is challenging due to the many-body character of the potential energy. Expansions in two- and three-body interactions may capture the physics [7], but are strictly valid only for weakly interacting systems. A large class of potentials, including the embedded atom method (EAM) [8], the bond order potentials [9], and the reactive FFs [10], share the physically motivated idea that the strength of a bond depends on the local environment, but the functional form of this dependence can only be given with crude approximations.

Machine learning (ML) methodologies are changing this state of affairs [11–20]. When trained on large data sets of atomic configurations and corresponding potential energies and forces, ML models can reproduce the original data accurately. In training these models, the atomic coordinates cannot be used as they appear in MD trajectories because their format does not preserve the translational, rotational, and permutational symmetry of the system. Different ML models address this issue in different ways. Two successful schemes are the Behler-Parrinello neural network (BPNN) [13] and the gradient-domain machine learning (GDML) method [19]. In the BPNN, symmetry is preserved by mapping the coordinates onto a large set of two- and three-body symmetry functions, which are, however, largely *ad hoc*. Fixing the symmetry functions may become painstaking in systems with many atomic species. In the GDML, the same goal is achieved by mapping the coordinates onto the eigenvalues of the Coulomb matrix, whose elements are the inverse distances between all distinct pairs of atoms. It is not straightforward how to use the Coulomb matrix in extended periodic systems.

So far, GDML has only been used for relatively small molecules.

In this Letter, we introduce a neural network (NN) based scheme for MD simulations, called deep potential molecular dynamics (DPMD), which overcomes the limitations associated with auxiliary quantities like the symmetry functions or the Coulomb matrix (All the examples presented in this work are tested using the DeePMD-kit package [21], which is available at [22]). In our scheme, a local reference frame and a local environment is assigned to each atom. Each environment contains a finite number of atoms, whose local coordinates are arranged in a symmetry preserving way following the prescription of the deep potential method [23], an approach that was devised to train a NN with the potential energy only. With typical AIMD data sets, this is insufficient to reproduce the trajectories. DPMD overcomes this limitation. In addition, the learning process in DPMD improves significantly over the deep potential method thanks to the introduction of a flexible family of loss functions. The NN potential constructed in this way reproduces accurately the AIMD trajectories, both classical and quantum (path integral), in extended and finite systems, at a cost that scales linearly with system size and is always several orders of magnitude lower than that of equivalent AIMD simulations.

In DPMD, the potential energy of each atomic configuration is a sum of "atomic energies" $E = \sum_i E_i$, where E_i is determined by the local environment of atom i within a cutoff radius R_c and can be seen as a realization of the embedded atom concept. The environmental dependence of E_i, which embodies the many-body character of the interactions, is complex and nonlinear. The NN is able to capture the analytical dependence of E_i on the coordinates of the atoms in the environment in terms of the composition of the sequence of mappings associated with the individual hidden layers. The additive form of E naturally preserves the extensive character of the potential energy. Because of the analyticity of the atomic energies, DPMD is, in principle, a conservative model.

E_i is constructed in two steps. First, a local coordinate frame is set up for every atom and its neighbors inside R_c [24]. This allows us to preserve the translational, rotational, and permutational symmetries of the environment, as shown in Fig. 1, which illustrates the format adopted for the local coordinate information $\{\boldsymbol{D}_{ij}\}$. The $1/R_{ij}$ factor present in $\boldsymbol{D}_{ij}$ reduces the weight of the particles that are more distant from atom i.

Next, $\{\boldsymbol{D}_{ij}\}$ serves as input of a deep neural network (DNN) [25], which returns E_i in output (Fig. 2). The DNN is a feed forward network, in which data flow from the input layer to the output layer (E_i), through multiple hidden layers consisting of several nodes that input the data d_l^{in} from the previous layer and output the data d_k^{out} to the next layer. A linear transformation is applied to the input data, i.e., $\tilde{d}_k = \sum_l w_{kl} d_l^{\text{in}} + b_k$, followed by action of a nonlinear function

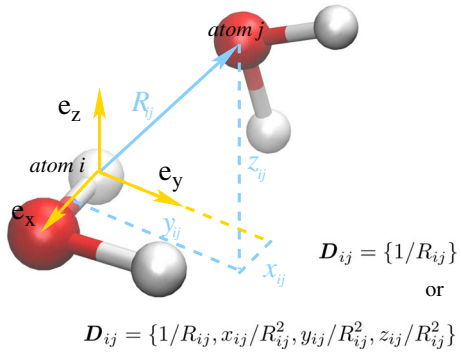

FIG. 1. Schematic plot of the neural network input for the environment of atom i, taking water as an example. Atom j is a generic neighbor of atom i, $(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)$ is the local frame of atom i, $\boldsymbol{e}_x$ is along the O—H bond, $\boldsymbol{e}_z$ is perpendicular to the plane of the water molecule, $\boldsymbol{e}_y$ is the cross product of $\boldsymbol{e}_z$ and $\boldsymbol{e}_x$, and (x_{ij}, y_{ij}, z_{ij}) are the Cartesian components of the vector $\boldsymbol{R}_{ij}$ in this local frame. R_{ij} is the length of $\boldsymbol{R}_{ij}$. The neural network input $\boldsymbol{D}_{ij}$ may either contain the full radial and angular information of atom j, i.e., $\boldsymbol{D}_{ij} = \{1/R_{ij}, x_{ij}/R_{ij}^2, y_{ij}/R_{ij}^2, z_{ij}/R_{ij}^2\}$ or only the radial information, i.e., $\boldsymbol{D}_{ij} = \{1/R_{ij}\}$. We first sort the neighbors of atom i according to their chemical species, e.g., oxygens first then hydrogens. Within each species, we sort the atoms according to their inverse distances to atom i, i.e., $1/R_{ij}$. We use $\{\boldsymbol{D}_{ij}\}$ to denote the sorted input data for atom i.

φ on $\tilde{d}_k$, i.e., $d_k^{\text{out}} = \varphi(\tilde{d}_k)$. In the final step from the last hidden layer to E_i, only the linear transformation is applied. The composition of the linear and nonlinear transformations introduced above provides the analytical representation of E_i in terms of the local coordinates. The technical details of this construction are discussed in the Supplemental Material [26]. In our applications, we adopt the hyperbolic tangent for φ and use five hidden layers with decreasing number of nodes per layer, i.e., 240, 120, 60, 30, and 10 nodes, respectively, from the innermost to the outermost layer. It is known empirically that the hidden layers greatly enhance the capability of neural networks to fit complex and highly

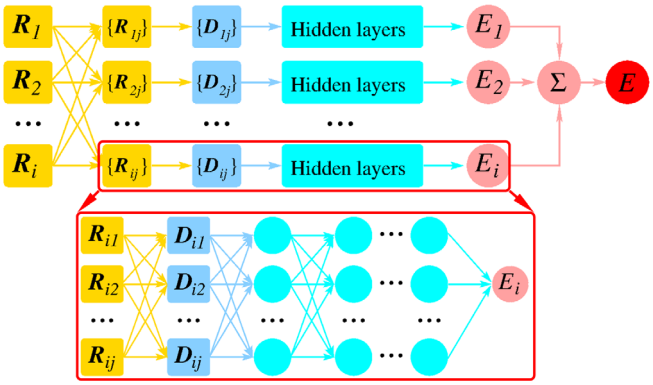

FIG. 2. Schematic plot of the DPMD model. The frame in the box is an enlargement of a DNN. The relative positions of all neighbors with respect to atom i, i.e., $\{\boldsymbol{R}_{ij}\}$, is first converted to $\{\boldsymbol{D}_{ij}\}$, then passed to the hidden layers to compute E_i.

nonlinear functional dependences [27,28]. In our case, only by including a few hidden layers could DPMD reproduce the trajectories with sufficient accuracy.

We use the Adam method [29] to optimize the parameters w_{kl} and b_k of each layer with the family of loss functions

$$L(p_\epsilon, p_f, p_\xi) = p_\epsilon \Delta\epsilon^2 + \frac{p_f}{3N}\sum_i |\Delta \boldsymbol{F}_i|^2 + \frac{p_\xi}{9}||\Delta\xi||^2. \quad (1)$$

Here Δ denotes the difference between the DPMD prediction and the training data, N is the number of atoms, ϵ is the energy per atom, $\boldsymbol{F}_i$ is the force on atom i, and ξ is the virial tensor $\Xi = -\frac{1}{2}\sum_i \boldsymbol{R}_i \otimes \boldsymbol{F}_i$ divided by N. In Eq. (1), p_ϵ, p_f, and p_ξ are tunable prefactors. When virial information is missing from the data, we set $p_\xi = 0$. In order to minimize the loss function in Eq. (1) in a well balanced way, we vary the magnitude of the prefactors during training. We progressively increase p_ϵ and p_ξ and decrease p_f, so that the force term dominates at the beginning, while energy and virial terms become important at the end. We find that this strategy is very effective and reduces the total training time to a few core hours in all the test cases.

To test the method, we have applied DPMD to extended and finite systems. As representative extended systems, we consider (a) liquid water at $P = 1$ bar and $T = 300$ K, at the path-integral AIMD (PI-AIMD) level, (b) ice Ih at $P = 1$ bar and $T = 273$ K, at the PI-AIMD level, (c) ice Ih at $P = 1$ bar and $T = 330$ K, at the classical AIMD level, and (d) ice Ih at $P = 2.13$ kbar and $T = 238$ K, which is the experimental triple point for ice I, II, and III, at the classical AIMD level. The variable periodic simulation cell contains 64 H_2O molecules in the case of liquid water and 96 H_2O molecules in the case of ices. We adopt $R_c = 6.0$ Å and use the full radial and angular information for the 16 oxygens and the 32 hydrogens closest to the atom at the origin, while retaining only radial information for all the other atoms within R_c. All the ice simulations include proton disorder. Deuterons replace protons in simulations (c) and (d). The hybrid version of Perdew-Burke-Ernzerhof (PBE0) + Tkatchenko-Scheffler (TS) [30,31] functional is adopted in all cases. As representative finite systems, we consider benzene, uracil, naphthalene, aspirin, salicylic acid, malonaldehyde, ethanol, and toluene, for which classical AIMD trajectories with the Perdew-Burke-Ernzerhof (PBE) + TS functional [31,32] are available [33]. In these systems, we set R_c large enough to include all the atoms and use the full radial and angular information in each local frame.

We discuss the performance of DPMD according to four criteria: (i) generality of the model; (ii) accuracy of the energy, forces, and virial tensor; (iii) faithfulness of the trajectories; and (iv) scalability and computational cost [34].

Generality.—Bulk and molecular systems exhibit different levels of complexity. The liquid water samples include quantum fluctuations. The organic molecules differ in composition and size, and the corresponding data sets include large numbers of conformations. Yet DPMD produces satisfactory results in all cases, using the same methodology, network structure, and optimization scheme. The excellent performance of DPMD in systems so diverse suggests that the method should be applicable to harder systems such as biological molecules, alloys, and liquid mixtures.

TABLE I. The RMSE of the DPMD prediction for water and ices in terms of the energy, the forces, and/or the virial. The RMSEs of the energy and the virial are normalized by the number of molecules in the system.

System	Energy (meV)	Force (meV/Å)	Virial (meV)
Liquid water	1.0	40.4	2.0
Ice Ih (b)	0.7	43.3	1.5
Ice Ih (c)	0.7	26.8	⋯
Ice Ih (d)	0.8	25.4	⋯

Accuracy.—We quantify the accuracy of energy, forces, and virial predictions in terms of the root-mean-square error (RMSE) in the case of water and ices (Table I) and in terms of the mean absolute error (MAE) in the case of the organic molecules (Table II). No virial information was used for the latter. In the water case, the RMSE of the forces is comparable to the accuracy of the minimization procedure in the original AIMD simulations, in which the allowed error in the forces was less than 10^{-3} Hartree/Bohr. In the case of the molecules, the predicted energy and forces are generally slightly better than the GDML benchmark.

MD trajectories.—In the case of water and ices, we perform path-integral or classical DPMD simulations at the thermodynamic conditions of the original models, using the I-PI software [35], but with much longer simulation time (300 ps). The average energy $\bar{E}$, density $\bar{\rho}$, radial distribution functions (RDFs), and a representative angular distribution function (ADF), i.e., a three-body correlation function, are reproduced with high accuracy. The results are summarized in Table III. The RDFs and ADF of the quantum trajectories of water are shown in Fig. 3. The

TABLE II. The MAE of the DPMD prediction for organic molecules in terms of the energy and the forces. The numbers in parentheses are the GDML results [19].

Molecule	Energy (meV)		Force (meV/Å)	
Benzene	2.8	(3.0)	7.6	(10.0)
Uracil	3.7	(4.0)	9.8	(10.4)
Naphthalene	4.1	(5.2)	7.1	(10.0)
Aspirin	8.7	(11.7)	19.1	(42.9)
Salicylic acid	4.6	(5.2)	10.9	(12.1)
Malonaldehyde	4.0	(6.9)	12.7	(34.7)
Ethanol	2.4	(6.5)	8.3	(34.3)
Toluene	3.7	(5.2)	8.5	(18.6)

TABLE III. The equilibrium energy and density, $\bar{E}$ and $\bar{\rho}$, of water and ices, with DPMD and AIMD. The numbers in square brackets are the AIMD results. The numbers in parentheses are statistical uncertainties in the last one or two digits. The training AIMD trajectories for the ices are shorter and more correlated than in the water case.

System	$\bar{E}$ (eV/H_2O)		$\bar{\rho}$ (g/m^3)	
Liquid water	−467.678(2)	[−467.679(6)]	1.013(5)	[1.013(20)]
Ice Ih (b)	−467.750(1)	[−467.747(4)]	0.967(1)	[0.966(6)]
Ice Ih (c)	−468.0478(3)	[−468.0557(16)]	0.950(1)	[0.949(2)]
Ice Ih (d)	−468.0942(2)	[−468.1026(9)]	0.986(1)	[0.985(2)]

RDFs of ice are reported in the Supplemental Material. A higher-order correlation function, the probability distribution function of the O—O bond orientation order parameter Q_6 [36], is additionally reported in the Supplemental Material and shows excellent agreement between DPMD and AIMD trajectories. In the case of the molecules, we perform DPMD at the same temperature of the original data, using a Langevin thermostat with a damping time $\tau = 0.1$ ps. The corresponding distributions of interatomic distances are very close to the original data (Fig. 4).

Scalability and computational cost.—All the physical quantities in DPMD are sums of local contributions. Thus, after training on a relatively small system, DPMD can be directly applied to much larger systems. The computational cost of DPMD scales linearly with the number of atoms. Moreover, DPMD can be easily parallelized due to its local decomposition and the near-neighbor dependence of its atomic energies. In Fig. 5, we compare the cost of DPMD fixed-cell simulations (NVT) of liquid water with that of equivalent simulations with AIMD and the empirical FF TIP3P (transferable intermolecular potential with 3 points) [41] in units of CPU core seconds/step/molecule.

While in principle the environmental dependence of E_i is analytical, in our implementation, discontinuities are present in the forces, due to adoption of a sharp cutoff radius, limitation of angular information to a fixed number of atoms, and abrupt changes in the atomic lists due to sorting. These discontinuities are similar in magnitude to those present in the AIMD forces due to finite numerical accuracy in the enforcement of the Born-Oppenheimer condition. In both cases, the discontinuities are much smaller than thermal fluctuations and perfect canonical evolution is achieved by coupling the systems to a thermostat. We further note that long-range Coulomb interactions are not treated explicitly in the current implementation, although implicitly present in the training data. Explicit treatment of Coulombic effects may be necessary in some applications and deserves further study.

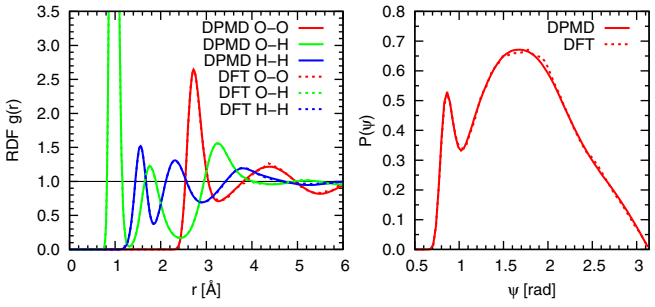

FIG. 3. Correlation functions of liquid water from DPMD and PI-AIMD. (Left) RDFs. (Right) The O-O-O ADF within a cutoff radius of 3.7 Å.

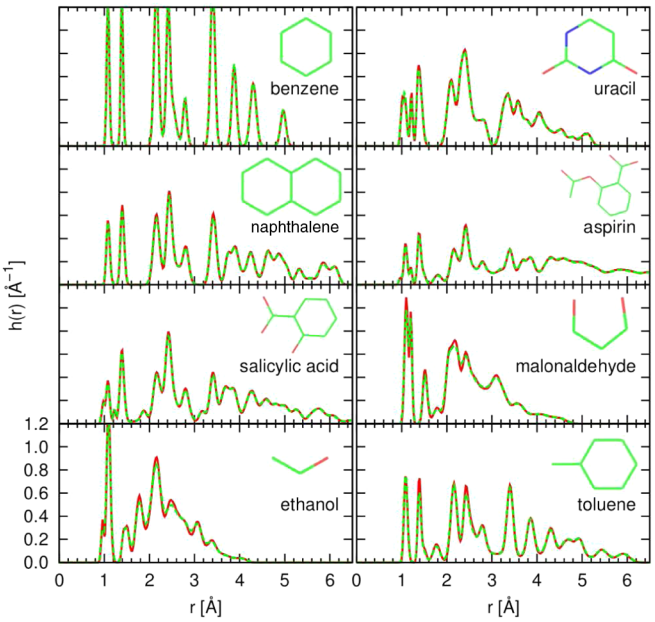

FIG. 4. Interatomic distance distributions of the organic molecules. The solid lines denote the DPMD results. The dashed lines denote the AIMD results.

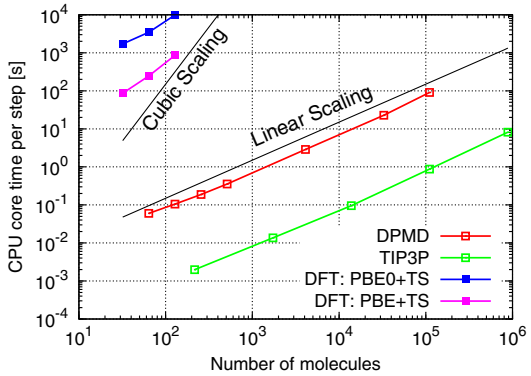

FIG. 5. Computational cost of MD step versus system size, with DPMD, TIP3P, PBE + TS, and PBE0 + TS. All simulations are performed on a Nersc Cori supercomputer with the Intel Xeon CPU E5-2698 v3. The TIP3P simulations use the `Gromacs` codes (version 4.6.7) [42]. The PBE + TS and PBE0 + TS simulations use the `Quantum Espresso` codes [43].

In conclusion, DPMD realizes a paradigm for molecular simulation, wherein accurate quantum mechanical data are faithfully parametrized by machine learning algorithms, which make possible simulations of DFT-based AIMD quality on much larger systems and for much longer time than with direct AIMD. While substantially more predictive than empirical FFs, DFT is not chemically accurate [44]. In principle, DPMD could be trained with chemically accurate data from high-level quantum chemistry [45] and/or quantum Monte Carlo calculations [46], but so far this has been prevented by the large computational cost of these calculations.

DPMD should also be very useful to coarse grain the atomic degrees of freedom, for example, by generating a NN model for a reduced set of degrees of freedom while using the full set of degrees of freedom for training. The above considerations suggest that DPMD should enhance considerably the realm of AIMD applications by successfully addressing the dilemma of accuracy versus efficiency that has confronted the molecular simulation community for a long time.

The authors acknowledge H.-Y. Ko and B. Santra for sharing the AIMD data on water and ice. The work of J. H. and W. E. is supported in part by Major Program of NNSFC under Grant No. 91130005, ONR Grant No. N00014-13-1-0338, DOE Awards No. DE-SC0008626 and No. DE-SC0009248, and NSFC Grant No. U1430237. The work of R. C. is supported in part by DOE-SciDAC Grant No. DE-SC0008626. The work of H. W. is supported by the National Science Foundation of China under Grants No. 11501039 and No. 91530322, the National Key Research and Development Program of China under Grants No. 2016YFB0201200 and No. 2016YFB0201203, and the Science Challenge Project No. JCKY2016212A502.

*wang_han@iapcm.ac.cn
†weinan@math.princeton.edu

[1] R. Car and M. Parrinello, Phys. Rev. Lett. **55**, 2471 (1985).
[2] D. Marx and J. Hutter, *Ab Initio Molecular Dynamics: Basic Theory and Advanced Methods* (Cambridge University Press, Cambridge, England, 2009).
[3] W. Kohn and L. J. Sham, Phys. Rev. **140**, A1133 (1965).
[4] K. Vanommeslaeghe, E. Hatcher, C. Acharya, S. Kundu, S. Zhong, J. Shim, E. Darian, O. Guvench, P. Lopes, I. Vorobyov, and A. Mackerell, Jr., J. Comput. Chem. **31**, 671 (2010).
[5] W. Jorgensen, D. Maxwell, and J. Tirado-Rives, J. Am. Chem. Soc. **118**, 11225 (1996).
[6] J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman, and D. A. Case, J. Comput. Chem. **25**, 1157 (2004).
[7] F. H. Stillinger and T. A. Weber, Phys. Rev. B **31**, 5262 (1985).
[8] M. S. Daw and M. I. Baskes, Phys. Rev. B **29**, 6443 (1984).
[9] D. W. Brenner, O. A. Shenderova, J. A. Harrison, S. J. Stuart, B. Ni, and S. B. Sinnott, J. Phys. Condens. Matter **14**, 783 (2002).
[10] A. C. T. van Duin, S. Dasgupta, F. Lorant, and W. A. Goddard, J. Phys. Chem. A **105**, 9396 (2001).
[11] A. P. Thompson, L. P. Swiler, C. R. Trott, S. M. Foiles, and G. J. Tucker, J. Comput. Phys. **285**, 316 (2015).
[12] T. D. Huan, R. Batra, J. Chapman, S. Krishnan, L. Chen, and R. Ramprasad, npj Comput. Mater. **3**, 37 (2017).
[13] J. Behler and M. Parrinello, Phys. Rev. Lett. **98**, 146401 (2007).
[14] J. Behler, J. Chem. Phys. **145**, 170901 (2016).
[15] T. Morawietz, A. Singraber, C. Dellago, and J. Behler, Proc. Natl. Acad. Sci. U.S.A. **113**, 8368 (2016).
[16] A. P. Bartók, M. C. Payne, R. Kondor, and G. Csányi, Phys. Rev. Lett. **104**, 136403 (2010).
[17] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, Phys. Rev. Lett. **108**, 058301 (2012).
[18] K. T. Schütt, F. Arbabzadah, S. Chmiela, K. R. Müller, and A. Tkatchenko, Nat. Commun. **8**, 13890 (2017).
[19] S. Chmiela, A. Tkatchenko, H. E. Sauceda, I. Poltavsky, K. T. Schütt, and K.-R. Müller, Sci. Adv. **3**, e1603015 (2017).
[20] J. S. Smith, O. Isayev, and A. E. Roitberg, Chem. Sci. **8**, 3192 (2017).
[21] Han Wang, Linfeng Zhan, Jiequn Han, and Weinan E, Comput. Phys. Commun. DOI: 10.1016/j.cpc.2018.03.016 (2018).
[22] https://github.com/deepmodeling/deepmd-kit.
[23] J. Han, L. Zhang, R. Car, and W. E, Commun. Comput. Phys. **23**, 629 (2018).
[24] Some flexibility can be used in the definition of the local frame of atom i. Usually we define it in terms of the two atoms closest to i, independent of their species. Exceptions to this rule are discussed in the Supplemental Material.
[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, Cambridge, MA, 2016).
[26] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.120.143001 for the technical details of the construction of the DNN model.
[27] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, in *Advances in Neural Information Processing Systems* (MIT Press, Cambridge, 2007), p. 153.
[28] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *Advances in Neural Information Processing Systems* (Curran, Red Hook, 2012), p. 1097.
[29] D. Kingma and J. Ba, arXiv:1412.6980.
[30] C. Adamo and V. Barone, J. Chem. Phys. **110**, 6158 (1999).
[31] A. Tkatchenko and M. Scheffler, Phys. Rev. Lett. **102**, 073005 (2009).
[32] J. P. Perdew, K. Burke, and M. Ernzerhof, Phys. Rev. Lett. **77**, 3865 (1996).
[33] http://quantum-machine.org/.
[34] We refer to the Supplemental Material for full details on the DPMD implementation and the training data sets, which includes [23,29,35–40].
[35] M. Ceriotti, J. More, and D. E. Manolopoulos, Comput. Phys. Commun. **185**, 1019 (2014).
[36] W. Lechner and C. Dellago, J. Chem. Phys. **129**, 114707 (2008).
[37] M. Ceriotti, D. E. Manolopoulos, and M. Parrinello, J. Chem. Phys. **134**, 084104 (2011).
[38] G. J. Martyna, M. L. Klein, and M. Tuckerman, J. Chem. Phys. **97**, 2635 (1992).

[39] M. Parrinello and A. Rahman, Phys. Rev. Lett. **45**, 1196 (1980).
[40] S. Ioffe and C. Szegedy, in *Proceedings of the 32nd International Conference on Machine Learning* (PMLR, 2015), Vol. 37, pp. 448–456.
[41] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein, J. Chem. Phys. **79**, 926 (1983).
[42] S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. Shirts, J. Smith, P. Kasson, D. van der Spoel, B. Hess, and E. Lindahl, Bioinformatics **29**, 845 (2013).
[43] P. Giannozzi *et al.*, J. Phys. Condens. Matter **29**, 465901 (2017).
[44] Conventionally, chemical accuracy corresponds to an error of 1 kcal/mol in the energy.
[45] J. D. Watts, J. Gauss, and R. J. Bartlett, J. Chem. Phys. **98**, 8718 (1993).
[46] D. Ceperley and B. Alder, Science **231**, 555 (1986).