MRS Communications (2019), 1 of 8

doi:10.1557/mrc.2019.80

MRS Communications

Research Letter

Artificial neural network correction for density-functional tight-binding molecular dynamics simulations

AQ2 **Junmian Zhu**, 1Grinnell College, Grinnell, IA, USA1
Van Quan Vuong, 2Bredesen Center for Interdisciplinary Research and Graduate Education, University of Tennessee, Knoxville, TN, USA2
Bobby G. Sumpter, 3Center for Nanophase Materials Sciences and Computational Sciences & Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA3
AQ1 **Stephan Irle**, 2Bredesen Center for Interdisciplinary Research and Graduate Education, University of Tennessee, Knoxville, TN, USA; 23Center for Nanophase Materials Sciences and Computational Sciences & Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA; 34Chemical Sciences Division, Oak Ridge National Laboratory, Oak Ridge, TN, USA4

Address all correspondence to Stephan Irle at **irles@ornl.gov**

(Received 18 February 2019; accepted 2 June 2019)

Abstract

The authors developed a Behler–Parrinello-type neural network (NN) to improve the density-functional tight-binding (DFTB) energy and force prediction. The Δ-machine learning approach was adopted and the NN was designed to predict the energy differences between the density functional theory (DFT) quantum chemical potential and DFTB for a given molecular structure. Most notably, the DFTB-NN method is capable of improving the energetics of intramolecular hydrogen bonds and torsional potentials without modifying the framework of DFTB itself. This improvement enables considerably larger simulations of complex chemical systems that currently could not easily been accomplished using DFT or higher level ab initio quantum chemistry methods alone.

Introduction

The density-functional tight-binding (DFTB) method[1–4] provides an efficient semi-empirical computational scheme to approximate density function theory (DFT) calculations. Using a minimal atomic orbital basis set for valence shell electrons and employing a two-center approximation for Hamiltonian matrix elements, the DFTB method efficiently reduces the computational cost of traditional DFT by 2–3 orders of magnitude.[5,6] This tremendous advance in computational speed can be exploited in long timescale molecular dynamics (MD) simulations of large-scale reactive processes.[6,7]

However, due to the approximations used in DFTB, in particular the use of the minimal atomic orbital basis set,[2] the representation of the electron density by a simplified point charge model,[1,4] and the neglect of three-center terms, traditional DFTB computed energies and gradients often deviate from DFT values.[6] For instance, the monopolar charge approximation and the use of a minimal basis set make it impossible for DFTB to describe out-of-plane polarization for planar molecular systems, to accurately capture atomic hybridization effects, etc. These deficiencies often result in problems for DFTB in the prediction of potential energy surfaces of highly polar systems.[6–8] Conventional DFTB parameterization methods often place emphasis on improving DFTB accuracy for the prediction of molecular structure and dynamics by adjusting the two-body repulsive potential.[9,10] However, the generally short-range two-body repulsive potential cannot address the neglect of high-order many-body effects by the DFTB formalism, which is out of reach by its very design. A major challenge is therefore to develop an efficient scheme to improve the accuracy of DFTB predictions without significantly increasing the computational cost of the method, or introducing unphysical new parameters.

In the past decade, neural networks (NNs) have been used as an efficient means to predict high-dimensional potential energy surfaces, most often only relying on training data for a given structural configuration space. Behler and Parinello constructed a generalized NN for fitting electronic energy based on radial and angular symmetry functions describing chemical environments of atoms in molecular or bulk systems.[11–14] Smith et al.[15] further modified the atomic environment descriptor for more accurate energy predictions by further allowing atom-type specifications. In addition, deep tensor NNs have also been developed for the representation of the atomic environment.[16]

In the present work, a Behler–Parinello-type NN (BPNN) is constructed to predict the deviation between DFTB and DFT-relative energies for a *given* molecular system, following Ramakrishnan's Δ-machine learning approach.[17] The idea of such a non-transferrable BPNN correction is that we can potentially harness the power of massively parallel computers by creating reference data for training NN corrections specifically for

the system under study while maintaining the simplicity yet physical underpinning of the tight-binding approach for production simulations. This relieves the strain on expected DFTB parameter accuracy as well as transferability and takes advantage of massively parallel (albeit embarrassingly parallel) quantum chemical computation, which usually scale poorly with system size.

We started our implementation using the BPNN Python code developed by Nguyen et al.,[18] which is available on Github. Our work represents a special case of the approach by Shen et al.,[19,20] who also employed a Behler–Parinello-type Δ-machine learning approach for their NN corrections in QM/MM simulations. We will make our code, which is written in Python and based on Keras/TensorFlow and ASE open-source program packages, available as an easy-to-use Δ-machine learning tool for general quantum chemical MD simulations. It is currently interfaced to the LGPL-licensed DFTB+ open-source code.[5]

Computational methodology

DFTB and NN methods

A detailed description of the DFTB formalism is given in Refs. 4 and 21. For the sake of brevity, we only introduce here essential traits of the DFTB family of approximations. The members of the hierarchy of DFTB methods are related to the order up to which the exact Kohn–Sham DFT energy of a molecular system $E[\rho]$ is expanded in a Taylor series around a reference electron density ρ^0 with respect to a difference electron density $\delta\rho$ as

$$E[\rho] = E^0[\rho^0] + E^1[\rho^0, \delta\rho] + E^2[\rho^0, (\delta\rho)^2] + E^3[\rho^0, (\delta\rho)^3] + \ldots, \quad (1)$$

where $\rho = \rho^0 + \delta\rho$.[1] The first-order DFTB (DFTB or DFTB1) method[2] includes only the first two terms in Eq. (6), the second-order DFTB [self-consistent charge DFTB (SCC-DFTB) or DFTB2] method[4] includes the first three terms, and the third-order DFTB (DFTB3) method[21] includes the first four terms.

In our formulation of the DFTB-NN methodology, we follow the work of Shen et al.[19,20] and utilize the difference between a given Kohn–-Sham DFT energy and the ordinary DFTB energy following the Δ-machine learning approach.[17] Note that we could, in principle, select any high-level method to define this difference. However, following the true spirit of DFTB, it is natural to utilize the NN to estimate the remainder of higher-order terms of the Taylor expansion in Eq. (6), as well as the effects of minimal basis set, two-center approximations, etc., as they can be thought of as introducing systematic errors that an NN should be easily able to spot as "patterns" in the data points it is presented. Consequently, the DFTB-NN energy prediction is simply the sum of the DFTB energy and the NN energy correction. For a given chemical system with a geometry "0" chosen as reference geometry, the DFT-relative energies $\Delta E_{\mathrm{DFT}}(X)$ for arbitrary geometries X are calculated as follows:

$$\Delta E_{\mathrm{DFT}}(X) = E_{\mathrm{DFT}}(X) - E_{\mathrm{DFT}}(0), \quad (2)$$

where $E_{\mathrm{DFT}}(X)$ and $E_{\mathrm{DFT}}(0)$ are DFT energies of geometry X and the reference geometry 0, respectively. Correspondingly, the DFTB relative energy $\Delta E_{\mathrm{DFTB}}(X)$ for a geometry X is calculated as follows:

$$\Delta E_{\mathrm{DFTB}}(X) = E_{\mathrm{DFTB}}(X) - E_{\mathrm{DFTB}}(0), \quad (3)$$

where $E_{\mathrm{DFTB}}(X)$ and $E_{\mathrm{DFTB}}(0)$ are DFTB energies of geometry X and the reference geometry "0". Both the geometry X and the reference geometry are the same for both DFT and DFTB calculations. The NN correction target energy for a geometry X is defined as the difference between DFT- and DFTB-relative energies:

$$E_{\mathrm{NN}}(X) = \Delta E_{\mathrm{DFT}}(X) - \Delta E_{\mathrm{DFTB}}(X). \quad (4)$$

The DFTB-NN energy for geometry X is then simply calculated as follows:

$$E_{\mathrm{DFTB-NN}}(X) = E_{\mathrm{DFTB}}(0) + \Delta E_{\mathrm{DFTB}}(X) + E_{\mathrm{NN}}(X). \quad (5)$$

Following Behler–Parrinello,[11] this "NN energy" E_{NN} for geometry X is a sum of atomic energy contributions

$$E_{\mathrm{NN}}(X) = \sum_{e=1}^{M} \sum_{j=1}^{N^e_{\mathrm{atom}}} E^e_j(X), \quad (6)$$

where E^e_j is the atomic energy of atom j of chemical element e, M is the number of chemical elements, and N^e_{atom} is the number of atoms belonging to element e the given system.

For each chemical element present in the molecular system, an atomistic feed-forward BPNN is built to predict the atomic energy of the atomic energy contribution E^e_j. For a high-dimensional NN potential, its output must be kept invariant with respect to permutation of the atomic index and to the rotation and translation of the input geometry to accurately represent the potential energy surface. Following Nguyen et al.'s approach,[18] we select two types of permutational, rotational, and translational invariant atomic number differentiated symmetry function vectors as the input of the NN. For each atom, a set of radial and angular symmetry function vectors are calculated from the neighboring atomic coordinates to describe its immediate chemical environment. The radial symmetry function vector $G^{\mathrm{Rad}}_{i,m}$ for atom i is a set of element-based contributions $G^{\mathrm{Rad}}_{i,m,Z}$ defined as follows:

$$G^{\mathrm{Rad}}_{i,m,Z} = \sum_{j \neq i}^{N^Z_{\mathrm{atom}}} e^{-\eta_m (R_{ij} - R_{s,m})^2}, \quad (7)$$

where j is the index of atoms with the same atomic number Z, R_{ij} is the distance between atom i and j, and the η_m and $R_{s,m}$ are

a set of arbitrary hyperparameters used in the mth symmetry function vector.[13] The angular symmetry function vector $G^{\mathrm{Ang}}_{i,m}$ for atom i is a set of element-based contribution $G^{\mathrm{Ang}}_{i,m,Z_1,Z_2}$ calculated as follows:

$$G^{\mathrm{Ang}}_{i,m,Z_1,Z_2} = \sum_{j\neq i}^{N^{Z_1}_{\mathrm{atom}}} \sum_{k\neq i,j}^{N^{Z_2}_{\mathrm{atom}}} \frac{2^{1-\zeta_m}(1+\lambda_m \cos\theta_{ijk})^{\zeta_m}}{e^{\eta_m(R^2_{ij}+R^2_{ik}+R^2_{jk})^2}}, \quad (8)$$

where j and k correspond to the index of atoms with the same atomic number $Z_1$and Z_2, R_{ik} is the distance between atom i and k, R_{jk} is the distance between atom j and k, θ_{ijk} is the angle formed by atoms i, j, and k, and ζ_m and λ_m are hyperparameters used in the mth symmetry function vector.[13] Similar to Smith et al.'s atomic environment vector,[15] only contributions from neighboring atoms of the same chemical element are summed into the same symmetry function vector in order to differentiate the atom type. In this proof-of-concept work, we do not introduce any cutoff for symmetry functions. The implementation of cutoff functions will be exploited in the future work.

A set of sub-NNs are built for all the elements present in the molecular system. The elemental sub-NNs use the symmetry function for the input and output of the atomic energy E^e_j in Eq. (6). Each feed-forward sub-NN consists of three layers with tan$h$ as an activation function and a linear layer to sum up all the node outputs into one atomic energy E^e_j. The output of node n in the mth layer is given by the following equation:

$$y^m_n = f^m_n\left(b^m_n + \sum_{p=1}^{N_{m-1}} a^{m-1,m*}_{p,n} y^{m-1}_n\right), \quad (9)$$

where y^m_n is the node output, f^m_n is the activation function, $a^{m-1,m}_{q,n}$ is the weight-connecting nodes between neighboring layers, and y^{m-1}_n is the output of the node in the previous layer. A detailed review of the BPNNs can be found in Refs.11 and 13.

A four-layer NN is constructed for each element from the open-source machine learning package Keras[22] using the TensorFlow backend.[23] The tan$h$ function is selected as the activation function for the first three layers and end with a linear layer to sum up into one atomic energy. Each layer consists of 34 nodes and the Nguyen–Widrow weight initialization[24] procedure was applied to initialize the weights for each elemental sub-NN.

In the DFTB-NN scheme, the force on atom i with respect to changes in a component of its Cartesian coordinate α is the sum of the ordinary DFTB force and the analytic NN gradient correction, obtained via a Jacobian matrix $\partial G_{j,m}/\partial\alpha$ as

$$F^{i,\alpha}_{\mathrm{DFTB-NN}} = F^{i,\alpha}_{\mathrm{DFTB}} - \sum_{j=1}^{N_{\mathrm{atom}}} \sum_{m=1}^{N_{\mathrm{sym}}} \frac{\partial E^e_j}{\partial G_{j,m}} \frac{\partial G_{j,m}}{\partial \alpha}. \quad (10)$$

The symbolic differentiation in TensorFlow was used to calculate the gradient of the NN energy with respect to each symmetry function vector. We employed the Adam optimizer implemented in the Keras python package to train the DFTB-NN model, minimizing the loss function f_{loss}, which is defined as follows:

$$f_{\mathrm{loss}} = \sqrt{\frac{1}{N}\sum_{X=1}^{L}(\Delta E_{\mathrm{DFTB}}(X) + E_{\mathrm{NN}}(X) - \Delta E_{\mathrm{DFT}}(X))^2}, \quad (11)$$

where L is the number of geometries in the training set, and $\Delta E_{\mathrm{DFTB}}(X)$ and $\Delta E_{\mathrm{NN}}(X)$ are the DFTB-relative energy and NN energy corrections for geometry X.

Computational details

As an illustrative example, the accuracy of the implemented DFTB-NN energy and the gradient was evaluated for the glycine molecule in both static and dynamic calculations. To simplify the discussion, we will only present data based on the DFTB2 method, using the MIO Slater–-Koster parameter set.[4] However, our implementation is also applicable to other members of the DFTB hierarchy of methods. As target high-level method, we selected the B3LYP functional[25] with Pople's 6-31G** basis set. For DFTB- and DFTB-NN MD simulations, we used the Velocity Verlet algorithm implemented in the Atomic Simulation Environment (ASE) library.[26] All DFT calculations were computed using the NWChem program.[27]

To construct a DFTB-NN model for the glycine molecule, a total of 276 symmetry function vectors are generated for each atom. Following Nguyen et al.'s approach,[18] the radial component uses 24 vectors for each element calculated by Eq. (7) with the $R_{s,m}$ evenly spread from 0.5 to 5 Å and $1/\sqrt{2\eta_m} = R_{s,m}$. The angular symmetry functions calculated from Eq. (8) adopt the parameters $\eta_m = [0.001, 0.01, 0.05]$ Å^{-1}, $\lambda_m = \pm 1$, and $\zeta_m = [1, 4, 6]$. To prevent overfitting, the early stopping method was employed and two dropout layers were applied to the second and third hidden layers of the elemental sub-NN. During the training process, the NN was continuously trained with a descending learning rate of $[5\times10^{-5}, 1.5\times10^{-5}]$ with corresponding epochs of [4000, 2000] and a decay rate of 1×10^{-5}.

Results and discussion

Accuracy of DFTB-NN-relative energies

Our choice for using glycine as a test case of the DFTB-NN approach was motivated by the fact that DFTB2 (and likewise DFTB3) tends to underestimate the strength of the O–H; · · · N hydrogen bond relative to that of the O–H; · · · O and N–H; · · · O hydrogen bond, as can be seen in Supplementary Fig. S1.

A training and two test sets were generated focusing on variations in the dihedral angles of the glycine molecule, followed by a short DFTB2 MD simulation. For the training set, 50 initial conformations were randomly generated using a modified version of the Kick^3 program[28] and followed by 100 steps of canonical ensemble (NVT) MD simulations at 300 K with

a step size of 1 fs. In total, 5000 structures and DFT energies were obtained for the training set. For the first test set named Glycine300K, 100 initial conformations were generated and followed by 1000 steps NVT MD simulations at 300 K. In total, 100,000 structures were obtained for the Glycine300K test set. The second test set named Glycine400K was generated in the same way as the first test set, but the NVT MD simulations were carried out at 400 K in order to generate geometries with higher variations than the training set. In total, we generated 200,000 DFT reference points for the test set. The size of the training set of glycine conformations was kept deliberately much smaller than that of the test set in order to avoid overfitting of the NN.

Figure 1 shows the comparison of DFTB-, DFTB-NN-, and DFT-relative energies ΔE for training, Glycine300K, and Glycine400K test sets, respectively. Overall, the DFTB-NN improves the accuracy of the DFTB-relative energy prediction. The DFTB method achieves a root mean square error (RMSE) of 2.39 and 2.54 kcal/mol for the Glycine300K and Glycine400K test sets, respectively. The DFTB-NN approach decreases the RMSE values significantly to 0.59 and 0.71 kcal/mol for the test sets.

Even though the DFTB-NN was trained using MD snapshots generated at 300 K with B3LYP ΔE in a range of [4.67, 35.80] kcal/mol, its RMSE for the Glycine400K test set, MD snapshots generated at 400 K with ΔE in the range of [2.89, 39.92] kcal/mol, is only 0.12 kcal/mol higher than that for the Glycine300K test set. The small difference in RMSE for the two test sets indicates that the trained DFTB-NN model is applicable within this temperature span. The NN is thus shown to be capable of learning from the relatively small training set and accurately predicts the high-dimensional potential energy surface.

Accuracy of the DFTB-NN gradient

In order to validate the implemented DFTB-NN analytic gradient, the difference between the analytic and numerical gradients was evaluated for a particular geometry of the glycine molecule. The maximum difference between the analytic and the numerical gradients was found to be only 6.0×10^{-6} hartree/bohr. Furthermore, the accuracy of the DFTB-NN gradient was evaluated by examining the energy conservation of DFTB-NN and traditional DFTB MD simulations in the microcanonical ensemble (NVE). NVE MD simulations were performed for 10 ps with the time step of 0.5 fs using velocities initiated at 300 K. As shown in Fig. 2, the DFTB-NN was able to keep the variation in total energy (kinetic MD energy plus total electronic + nuclear energy as potential energy) during the MD simulation lower than ± 0.03 kcal/mol, which is at the same level as DFTB. Ideally, the total energy needs to be perfectly conserved in such NVE simulations, and the deviation of actual total energy from 0 is a good measure of the accuracy of the MD simulation. Figure 2 also shows that the mean of energy fluctuation is very close to zero and of the same magnitude for both DFTB and DFTB-NN, indicating the good accuracy of the DFTB-NN force calculation.

To demonstrate the computational performance of the implemented DFTB-NN, we compared the wall-clock time required to carry out a single-point energy + gradient calculation for the glycine molecule. On a single Xeon 3.00 GHz (E3-1505M) CPU core, the calculation took 0.256 s with DFTB2 and 0.302 s with DFTB2-NN. The wall-clock time was averaged over 1000 steps of a MD trajectory.

Potential energy profile of the N–C–O–H dihedral angle

DFTB neglects three-center, four-center, and higher-order many-body contributions as well as multipole interactions, which particularly affect the description of the chemical environment related to dihedral angles and (intramolecular) hydrogen bonds.[8] To examine the accuracy of our trained DFTB-NN model, the DFTB-NN potential energy profile for a 360° rotation of the N–C–O–H dihedral angle of the glycine molecule was compared to the corresponding PES computed by B3LYP. At first, the glycine geometry was optimized using the B3LYP/6-31G** method, following a partially relaxed potential energy surface scan at the same level of theory for the N–C–O–H dihedral angle of the molecule. The corresponding energies for each geometry along the partially relaxed energy scan were calculated at the DFTB2 and DFTB2-NN levels of theory. The only other frozen parameter during the relaxed scan was the N–C–C–O dihedral angle, which we fixed at −7.0 in order to avoid sudden jumps between different conformations along the energy profile One sudden geometrical change remains for which the NH_2 group suddenly changes its orientation, see Fig. 3 (small insets at 282.3). The geometries of this energy scan are given in Supplementary Material.

Figure 3 illustrates that DFTB2 incorrectly predicts the global minimum at the position of the N–H; · · · N intramolecular hydrogen bond, whereas DFT predicts, in agreement with other higher levels of theory, that the true global minimum should be located at the position of the O–H; · · · N bond. As a consequence, DFTB2 predicts incorrectly that the *cis*-isomer of glycine with N–C–O–H dihedral angle of 0° is less stable than the *trans*-isomer of 180°. DFT, on the other hand, predicts that the *cis*-isomer is lower in energy. Clearly, DFTB2 fails to yield accurate prediction, because it uses Coulombic interactions between atomic point charges that cannot accurately model the hydrogen atom interactions.[7] As a remedy, the DFTB-NN is capable of reproducing the DFT PES with a maximum absolute error of 1.7 kcal/mol (9.4 kcal/mol in the case of DFTB) and predicts that the *cis*-isomer is more stable. It is apparent that DFTB-NN corrects faithfully flaws in the DFTB hydrogen bonding energies and approaches the accuracy of DFT without modifying the DFTB formalism itself, as for instance by the use of hydrogen corrections.[29,30]

Fig. 1 - B/W online

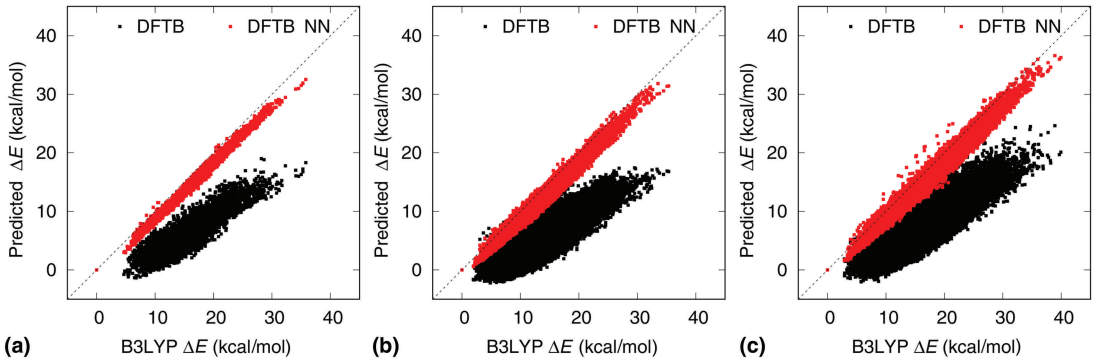

Figure 1. Comparison of DFTB and DFTB-NN ΔE to DFT ΔE for (a) the training set, (b) the Glycine300K test set, and (c) the Glycine400K test set.

Energy landscape of glycine

To further evaluate the accuracy of DFTB-NN for the description of intramolecular hydrogen bonds and the dihedral angle, we examined the potential energy landscape of the C–C–O–H and N–C–C–O dihedral angles using all geometries of the Glycine300K test set. Figure 4 shows the relative energy of DFTB, DFTB-NN, and B3LYP varying with the two dihedral angles. The scale of ΔE is mapped according to the color scale shown in the plot.

DFTB systematically underestimates the potential energy when compared with DFT. Overall, the DFTB-NN energy landscape matches the B3LYP energy landscape precisely. Remaining deviations exist for O–C–C–H dihedral angles in the range of [0, 60] and [90, 120], although the magnitude of these deviations is relatively small.

Conclusions

We successfully developed and implemented a BPNN for correcting the DFTB energy calculation to match higher-level quantum chemical potential energy surfaces for the purpose of running MD simulations. We adopted Ramakrishnan's Δ-machine learning approach,[17] following the approach recently published for correcting QM/MM calculations.[19,20] We found that the newly developed DFTB-NN was capable of reducing the RMSE of the DFTB energy calculation against DFT below 1.0 kcal/mol and therefore improved its accuracy. The inclusion of the two-body radial and three-body angular symmetry functions as input for DFTB-NN facilitates its ability to interpolate the energy differences between DFT and DFTB. Most notably, the DFTB-NN method is capable of improving the energetics of intramolecular hydrogen bonds and torsional potentials without having to modify the framework of the DFTB method itself. This improvement is adequate to amplify the future applications of DFTB to highly polar systems and for sizes not approachable by DFT alone. Our code is available upon request from the authors and will be published on an online repository in the near future.

Notice of copyright

Fig. 2 - B/W online

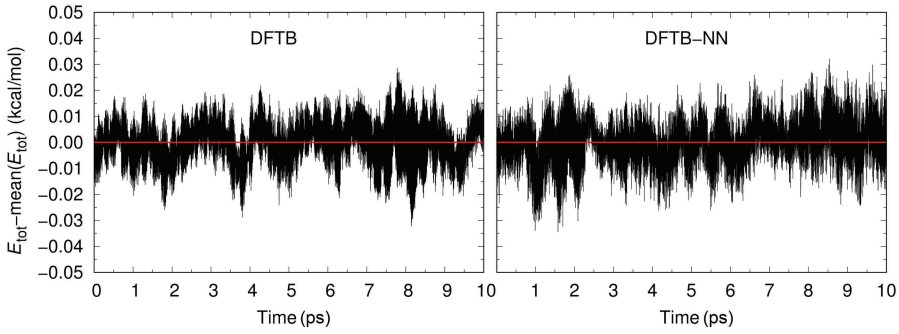

Figure 2. DFTB and DFTB-NN total energy of the glycine molecule fluctuate as a function of NVE simulation time.

Fig. 3 - B/W online

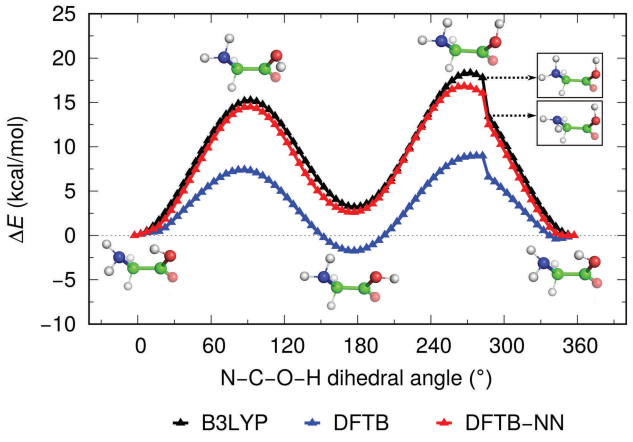

Figure 3. Potential energy profile for DFTB2 and DFTB2-NN in comparison to B3LYP/6-31G** as a function of the N-C-O-H dihedral angle in glycine.

sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan).

Acknowledgments

The authors thank Dr. Andreas Goetz (SDSC) and Dr. Thuong Nguyen (UCSD) for providing us their initial BPNN code for water clusters. We also thank Dr Jacek Jakowski, Dr. Panchapakesan Ganesh, and Dr. Maxim Ziatdinov (all ORNL) for helpful discussions. J.Z. acknowledges support under the Oak Ridge Science Semester (ORSS) grant from the Oak Ridge Institute for Science and Education (ORISE). V.Q.V. acknowledges support by an Energy Science and Engineering Fellowship of the Bredesen Center for Interdisciplinary Research and Graduate Education at the University of Tennessee, Knoxville. S.I. acknowledges support by the Fluid Interface Reactions, Structures and Transport (FIRST) Center, an Energy Frontier Research Center funded by the U.S. DOE Office of Science. Calculations were performed at the Center for Nanophase Materials Sciences, which is a U.S. Department of Energy Office of Science User Facility.

Supplementary material

The supplementary material for this article can be found at https://doi.org/10.1557/mrc.2019.80.

References

1 W.M.C. Foulkes and R. Haydock: Tight-binding models and density-functional theory. *Phys. Rev. B* **39**, 12520–12536 (1989).
2 D. Porezag, T. Frauenheim, T. Köhler, G. Seifert, and R. Kaschner: Construction of tight-binding-like potentials on the basis of density-functional theory: application to carbon. *Phys. Rev. B* **51**, 12947–12957 (1995).
3 G. Seifert, D. Porezag, and T. Frauenheim: Calculations of molecules, clusters, and solids with a simplified LCAO-DFT-LDA scheme. *Int. J. Quantum Chem.* **58**, 185–192 (1996).
4 M. Elstner, D. Porezag, G. Jungnickel, J. Elsner, M. Haugk, T. Frauenheim, S. Suhai, and G. Seifert: Self-consistent-charge density-functional tight-binding method for simulations of complex materials properties. *Phys. Rev. B* **58**, 7260–7268 (1998).
5 B. Aradi, B. Hourahine, and T. Frauenheim: DFTB+, a sparse matrix-based implementation of the DFTB method. *J. Phys. Chem. A* **111**, 5678–5684 (2007).
6 M. Elstner and G. Seifert: Density functional tight binding. *Philos. Trans. R. Soc. A* **372**, 20120483–20120494 (2014).
7 A.S. Christensen, T. Kubar, Q. Cui and M. Elstner: Semiempirical quantum mechanical methods for noncovalent interactions for chemical and biochemical applications. *Chem. Rev.* **116**, 5301–5337 (2016).
8 K.H. Lee, U. Schnupf, B.G. Sumpter, and S. Irle: Performance of density-functional tight-binding in comparison to ab initio and first-principles methods for isomer geometries and energies of glucose epimers in vacuo and solution. *ACS Omega* **3**, 16899–16915 (2018).
9 M. Gaus, A. Goez, and M. Elstner: Parametrization and benchmark of DFTB3 for organic molecules. *J. Chem. Theory Comput.* **9**, 338–354 (2013).
10 V.Q. Vuong, J. Akkarapattiakal Kuriappan, M. Kubillus, J.J. Kranz, T. Mast, T.A. Niehaus, S. Irle, and M. Elstner: Parametrization and benchmark of long-range corrected DFTB2 for organic molecules. *J. Chem. Theory Comput.* **14**, 115–125 (2018).
11 J. Behler and M. Parrinello: Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **98**, 146401 (2007).
12 J. Behler: Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *J. Chem. Phys.* **134**, 074106 (2011).
13 J. Behler: Representing potential energy surfaces by high-dimensional neural network potentials. *J. Phys.: Condens. Matter* **26**, 183001 (2014).
14 J. Behler: Constructing high-dimensional neural network potentials: a tutorial review. *Int. J. Quantum Chem.* **115**, 1032–1050 (2015).
15 J.S. Smith, O. Isayev, and A.E. Roitberg: ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chem. Sci.* **8**, 3192–3203 (2017).
16 K.T. Schütt, F. Arbabzadah, S. Chmiela, K.R. Müller, and A. Tkatchenko: Quantum-chemical insights from deep tensor neural networks. *Nat. Commun.* **8**, 13890 (2017).
17 R. Ramakrishnan, P.O. Dral, M. Rupp, and O.A. von Lilienfeld: Big data meets quantum chemistry approximations: the Δ-machine learning approach. *J. Chem. Theory Comput.* **11**, 2087–2096 (2015).
18 T.T. Nguyen, E. Székely, G. Imbalzano, J. Behler, G. Csányi, M. Ceriotti, A. W. Götz, and F. Paesani: Comparison of permutationally invariant polynomials, neural networks, and Gaussian approximation potentials in representing water interactions through many-body expansions. *J. Chem. Phys.* **148**, 241725 (2018).
19 L. Shen, J. Wu, and W. Yang: Multiscale quantum mechanics/molecular mechanics simulations with neural networks. *J. Chem. Theory Comput.* **12**, 4934–4946 (2016).
20 L. Shen and W. Yang: Molecular dynamics simulations with quantum mechanics/molecular mechanics and adaptive neural networks. *J. Chem. Theory Comput.* **14**, 1442–1455 (2018).
21 M. Gaus, Q. Cui, and M. Elstner: DFTB3: extension of the self-consistent-charge density-functional tight-binding method (SCC-DFTB). *J. Chem. Theory Comput.* **7**, 931–948 (2011).
22 Keras: Deep Learning for humans. https://github.com/keras-team/keras.
23 M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D.G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng: TensorFlow: a system for large-scale machine learning.In *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, (2016); pp. 265–283.
24 D. Nguyen and B. Widrow: Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights. In 1990 IJCNN International Joint Conference on Neural Networks, (1990); pp. 21–26.
25 A.D. Becke: Density-functional thermochemistry. III. The role of exact exchange. *J. Chem. Phys.* **98**, 5648–5652 (1993).

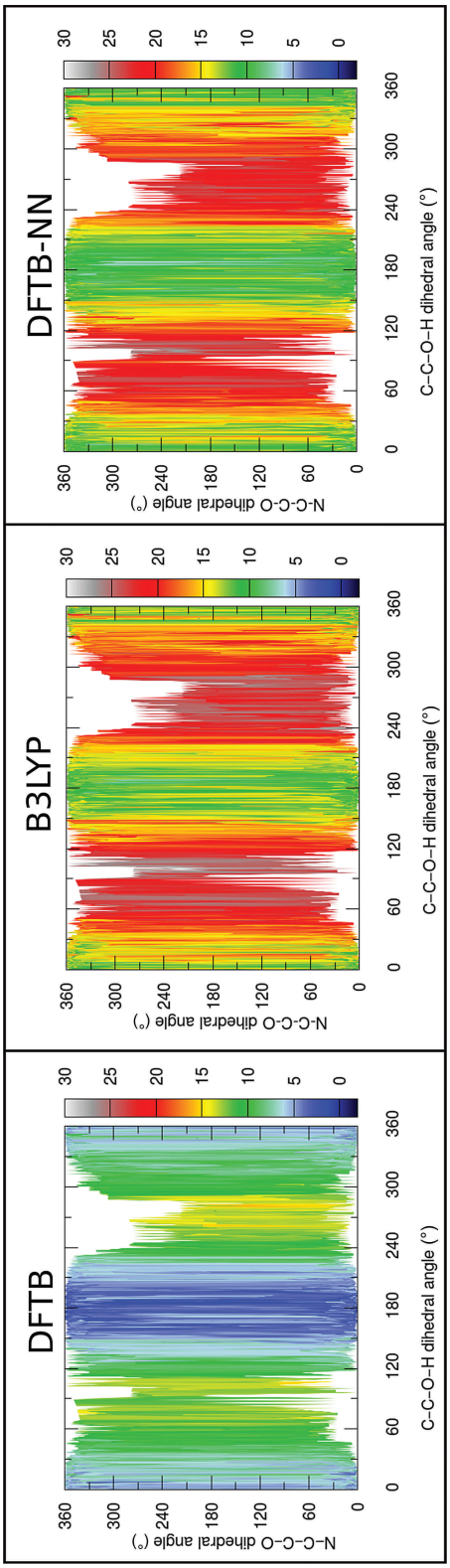

Figure 4. DFTB-, DFT-, and DFTB-NN-relative energies as a function of C–C–O–H and N–C–C–O dihedral angles of glycine.

Fig. 4 - B/W online

26 A. Hjorth Larsen, J. Jørgen Mortensen, J. Blomqvist, I.E. Castelli, R. Christensen, M. Dułak, J. Friis, M.N. Groves, 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: The atomic simulation environment-a Python library for working with atoms. *J. Phys.: Condens. Matter* **29**, 273002 (2017).
27 M. Valiev, E. Bylaska, N. Govind, K. Kowalski, T. Straatsma, H. Van Dam, D. Wang, J. Nieplocha, E. Apra, T. Windus, and W. de Jong: NWChem: a comprehensive and scalable open-source solution for large scale molecular simulations. *Comput. Phys. Commun.* **181**, 1477–1489 (2010).
28 M.A. Addicoat, S. Fukuoka, A.J. Page, and S. Irle: Stochastic structure determination for conformationally flexible heterogenous molecular clusters: application to ionic liquids. *J. Comput. Chem.* **34**, 2591–2600 (2013).
29 J. Rezáč, K.E. Riley, and P. Hobza: S66: a well-balanced database of benchmark interaction energies relevant to biomolecular structures. *J. Chem. Theory Comput.* **7**, 2427–2438 (2011).
30 J. Rezáč: Empirical self-consistent correction for the description of hydrogen bonds in DFTB3. *J. Chem. Theory Comput.* **13**, 4804–4817 (2017).