JCTC Journal of Chemical Theory and Computation

pubs.acs.org/JCTC Article

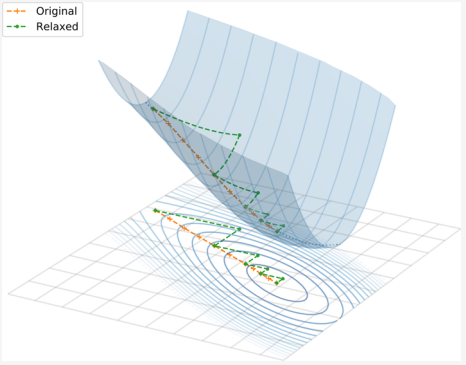
# DeePKS: A Comprehensive Data-Driven Approach toward Chemically Accurate Density Functional Theory

Yixiao Chen, Linfeng Zhang,* Han Wang,* and Weinan E

Cite This: J. Chem. Theory Comput. 2021, 17, 170−181 Read Online

ACCESS | Metrics & More | Article Recommendations

**ABSTRACT:** We propose a general machine learning-based framework for building an accurate and widely applicable energy functional within the framework of generalized Kohn−Sham density functional theory. To this end, we develop a way of training self-consistent models that are capable of taking large datasets from different systems and different kinds of labels. We demonstrate that the functional that results from this training procedure gives chemically accurate predictions on energy, force, dipole, and electron density for a large class of molecules. It can be continuously improved when more and more data are available.






## 1. INTRODUCTION

Predicting the ground-state information of a many-electron system in an environment of clamped ions is a fundamental task in the field of molecular modeling. Over the past few decades, a wide variety of methods have been developed for addressing this problem, such as quantum Monte Carlo, post-Hartree−Fock (HF) methods (also known as wave function theory, WFT), density functional theory (DFT),[1] etc. In general, these methods follow a well-known trade-off between accuracy and efficiency. The cost of exact WFT methods like full configuration interaction (FCI)[2] usually scales exponentially with system size. Coupled-cluster singles, doubles, and perturbative triples (CCSD(T)),[3] the method often referred to as the golden standard of quantum chemistry, has a cost that scales as $O(N^7)$ with respect to the number of electrons $N$. The cost of Kohn−Sham (KS) DFT[4] and its generalized version[5] typically scales as $O(N^3{\sim}N^4)$. However, currently available DFT models, although much more efficient, are much less accurate compared with FCI and CCSD(T) due to the approximate nature of the functionals involved.

Developing accurate and efficient DFT functionals is among the world's hardest and most important parameter fitting problems. As for all parameter fitting problems, we need a functional form with some free parameters and a way to optimize these parameters. The key notion in this context is universality. In principle, the DFT functionals are universal and we would like our approximate functionals to be as universal as possible.

It should be noted immediately that truly universal and computationally efficient functionals are very difficult, if not impossible, to come by. Therefore, our goal should be to develop a functional that is efficient and chemically accurate for all the systems that can be reasonably represented by the data available.

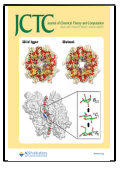
To this end, we look for models with the following requirements in mind:

1. We need to have a functional form (for these approximate functionals) that is expressive enough so that the behavior of different systems, whether small or large molecules or condensed systems, can all be accommodated.
2. We should also make maximum use of existing high-quality data, including data for different systems and data with different kinds of labels, such as energy, force, and electron density. The model should be continuously improvable as more and more data become available.

Since condensed systems involve other nontrivial technical issues, we choose to focus first on molecules. For a similar reason, we do not discuss the analytical conditions that are used in the so-called nonempirical functionals.[6,7] Most of these conditions are derived in some limiting cases, such as uniform electron gas. They are less relevant to molecules or the generalized Kohn−Sham scheme that we are going to use.

A reasonably successful (non-self-consistent) model that accomplishes the first requirement, termed deep post-Hartree−

Received: August 21, 2020
Published: December 9, 2020






https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

Fock (DeePHF), has been developed in ref 8 for molecules. By exploiting both physical constraints from symmetries and the unprecedented expressivity of neural network (NN) functions, DeePHF succeeded in achieving chemical accuracy for the energy at a cost comparable to Hartree−Fock (HF). It has demonstrated impressive performance on existing datasets for molecules. One main objective of the current work is to extend DeePHF to a self-consistent framework such as KS-DFT. We will adopt the generalized Kohn−Sham (GKS) formalism, with the domain of our functional elevated from pure density to Kohn−Sham orbitals, so that the functional space represented is much larger. At the same time, we will make sure that the second requirement listed above is also fulfilled.

Despite several earlier attempts,[9−14] there have been serious difficulties involved in this task. For machine learning-based models such as DeePHF, it was the gradient-based optimization schemes that make them efficiently trainable. Gradient-based methods can hardly be used in the self-consistent framework since it is very expensive to compute the gradients of the self-consistent energy, force, and density, with respect to the NN parameters. For this reason, an earlier attempt reported in ref 14 used Monte Carlo, a gradient-free optimization scheme. This is prohibitively expensive in the self-consistent setup, particular with large datasets. When the training data are limited to only the energies of a few molecules, the pioneering work reported in ref 11 successfully developed a gradient-based strategy by effectively decoupling the self-consistent constraint and the gradient-based training. We will follow a similar strategy, but we have to develop a modified reformulation to make the process more efficient so that much larger datasets can be handled. When the training data also include alternative labels other than energy, such as forces and electron density, to the best of our knowledge, no effective gradient-based method has been developed. We will present a new training scheme that overcomes these difficulties in a very elegant way.

We name the approach proposed here deep Kohn−Sham (DeePKS) to highlight the self-consistent nature that distinguishes this method from our previous work. Self-consistency enables calculating force and density-related properties naturally from DeePKS, a key feature that differs from a pure energy model. We also use DeePKS to refer to the model (i.e., functionals) obtained this way. DeePKS obeys all physical and gauge symmetries and is consistent with all known high-quality data. In addition, it can be continuously improved as more and more data become available. We also note that the training schemes developed here can be used in other situations when some self-consistent models are trained.

## 2. METHODS

**2.1. (Generalized) Kohn−Sham Theory.** We first give a brief overview of the (generalized) Kohn−Sham theory. We start from the many-body Schrödinger equation of $N$ electrons indexed by $i$

$$(T + W + V_{\text{ext}})\Psi(x_1, x_2, ..., x_N) = E_{\text{tot}}\Psi(x_1, x_2, ..., x_N) \tag{1}$$

where we use $E_{\text{tot}}$ to denote the ground-state energy of the $N$-electron Schrödinger equation. Here, $T = -\frac{1}{2}\nabla^2$ and $W = \frac{1}{2}\sum_{i,j}\frac{1}{|x_i - x_j|}$ denote the kinetic operator and electron−electron interactions, respectively. $V_{\text{ext}}$ stands for the external potential. For example, in an atomic system with $M$ ions indexed by $I$, $V_{\text{ext}} = \sum_i \mathcal{V}_{\text{ext}}(x_i) = \sum_{I,i}\frac{Z_I}{|x_I - x_i|}$.

Following the variational principle, the ground-state energy can also be written as

$$\begin{aligned} E_{\text{tot}} &= \min_{\Psi}\langle\Psi|T + W + V_{\text{ext}}|\Psi\rangle \\ &= \min_{\Psi}\{G_0[\Psi] + E_{\text{ext}}[\rho[\Psi]]\} \end{aligned} \tag{2}$$

where

$$G_0[\Psi] = \langle\Psi|T + W|\Psi\rangle \tag{3}$$

$$E_{\text{ext}}[\rho] = \int \mathrm{d}x\, \mathcal{V}_{\text{ext}}(x)\rho(x) \tag{4}$$

According to the well-known Hohenberg−Kohn theorem,[1] this problem is equivalent to another minimization problem with respect to the electron density $\rho$

$$E_{\text{tot}} = \min_{\rho(x)\to N}\{F_{\text{HK}}[\rho] + E_{\text{ext}}[\rho]\} \tag{5}$$

$$F_{\text{HK}}[\rho] = \min_{\Psi\to\rho(x)} G_0[\Psi] \equiv \min_{\Psi\to\rho(x)}\langle\Psi|T + W|\Psi\rangle \tag{6}$$

Equation 6 defines the Hohenberg−Kohn (HK) functional $F_{\text{HK}}[\rho]$ using the Levy−Lieb constrained search formulation.[15,16] Note here that both $F_{\text{HK}}[\rho]$ and $G_0[\Psi]$ are considered to be universal, meaning that they do not depend explicitly on the external potential $V_{\text{ext}}$.

Directly solving the ground-state energy or representing the HK functional can be very difficult since it involves dealing with the $N$-particle wave function. Therefore, one often resorts to the popular Kohn−Sham (KS) scheme to simplify this problem. The key ingredient of KS-like theories is to replace the general $N$-particle ground-state $\Psi$ with a model system, whose ground state can be represented by a single Slater determinant $\Phi = \frac{1}{\sqrt{N!}}\det[\varphi_i(x_j)]$, where we use $\{\varphi_i\}$ to denote a set of orthonormal single-particle orbitals. The energy functional can also be written as $G[\Phi] = G[\{\varphi_i\}]$. As a result, the ground-state energy $E_{\text{KS}}$ and density functional $F_{\text{KS}}$ are given by

$$E_{\text{KS}} = \min_{\rho(x)\to N}\{F_{\text{KS}}[\rho] + E_{\text{ext}}[\rho]\} \tag{7}$$

$$F_{\text{KS}}[\rho] = \min_{\Phi\to\rho(x)} G[\Phi] = \min_{\substack{\{\varphi_i\}\to\rho(x)\\ \langle\varphi_i|\varphi_j\rangle=\delta_{ij}}} G[\{\varphi_i\}] \tag{8}$$

Depending on how the functional $G[\Phi]$ is chosen, the above formulation gives many different theories. To name a few:

- If we leave $G$ unchanged from $G_0$, then we get the Hartree−Fock theory

$$\begin{aligned} G_{\text{HF}}[\Phi] &= G_0[\Phi] = \langle\Phi|T + W|\Phi\rangle \\ &= \langle\Phi|T|\Phi\rangle + E_{\text{H}}[\rho] + E_{\text{F}}[\{\varphi_i\}] \end{aligned} \tag{9}$$

  where $E_{\text{H}}[\rho]$ and $E_{\text{F}}[\{\varphi_i\}]$ denote the Coulomb (Hartree) energy and exchange (Fock) energy, respectively. Note here that $E_{\text{H}}$ depends only on the electron density $\rho(x) = \sum_i|\varphi_i(x)|^2$.
- If we constrain $G$ such that the only term that explicitly depends on $\Phi$ is the kinetic energy, then we get the standard KS theory

$$G_{\text{KS}}[\Phi] = \langle\Phi|T|\Phi\rangle + E_{\text{H}}[\rho] + E_{\text{xc}}[\rho] \tag{10}$$


https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

where $E_{xc}[\rho]$ is the so-called exchange-correlation functional. Usually, $E_{xc}[\rho]$ can be split into two parts, the exchange energy $E_x[\rho]$ and the correlation energy $E_c[\rho]$.

- If we include part of the Fock exchange operator in addition to the standard exchange-correlation functional, then we get a standard version the hybrid Kohn−Sham theory

$$G_{\text{Hyb}}[\Phi] = \langle\Phi|T|\Phi\rangle + E_H[\rho] + \lambda E_F[\{\varphi_i\}] + (1-\lambda)E_x[\rho] + E_c[\rho] \tag{11}$$

where $\lambda$ is a tunable factor deciding how much the exact exchange operator is used.

The term generalized Kohn−Sham (GKS) theory simply refers to any choice of $G$ that does not satisfy the standard KS condition (eq 10). Many functionals fall into this class, including all hybrid functionals and most meta-GGA functionals.

A KS-like theory is considered to be exact if its choice of $G$ yields the same density functional as the original Hohenberg−Kohn functional, namely

$$F_{\text{KS}}[\rho] = F_{\text{HK}}[\rho] \tag{12}$$

Therefore, an exact theory would give the exact ground-state energy, $E_{\text{KS}} = E_{\text{tot}}$, as well as the exact ground-state density $\rho$. As an example, the aforementioned Hartree−Fock theory is obviously not exact. It remains an open question whether there exists a possible choice of $G$ in general that yields the exact functional and, hence, the exact ground-state density. In the context of standard KS theory, it is termed the problem of non-interacting $v$-representability. From this point of view, the GKS theory is at least as exact as the standard KS theory.

To solve the KS-like problem, we reformulate eq 7 as a direct minimization problem with respect to the single-particle orbitals $\{\varphi_i\}$, namely

$$E_{\text{KS}} = \min_{\{\varphi_i\},\langle\varphi_i|\varphi_j\rangle=\delta_{ij}} \{G[\{\varphi_i\}] + E_{\text{ext}}[\rho[\{\varphi_i\}]]\} \tag{13}$$

We now further require that the functional derivative of $G[\{\varphi_i\}]$ can be cast into the form of a single-particle operator

$$\frac{\delta G[\{\varphi_j\}]}{\delta\langle\varphi_i|} = O[\{\varphi_j\}]|\varphi_i\rangle \tag{14}$$

Therefore, using Lagrange multipliers on eq 13, we obtain the self-consistent field (SCF) equation

$$\mathcal{H}[\{\varphi_j\}]|\varphi_i\rangle \equiv (O[\{\varphi_j\}] + \mathcal{V}_{\text{ext}})|\varphi_i\rangle = \varepsilon_i|\varphi_i\rangle \quad \text{for} \quad i = 1...N \tag{15}$$

where we use H to denote the single-particle Hamiltonian. As an example, for the HF theory (eq 9), we have

$$O_{\text{HF}}[\{\varphi_j\}] = \mathcal{T} + \mathcal{V}_H[\rho] + \mathcal{V}_F[\{\varphi_j\}] \tag{16}$$

For the standard KS theory (eq 10), we have

$$O_{\text{KS}}[\{\varphi_j\}] = \mathcal{T} + \mathcal{V}_H[\rho] + \mathcal{V}_{xc}[\rho] \tag{17}$$

Here, we use $\mathcal{T}$, $\mathcal{V}_H$, $\mathcal{V}_F$, and $\mathcal{V}_{xc}$ to denote single-particle kinetic, Coulomb, exact exchange, and exchange-correlation operators, respectively.

**2.2. Model Construction.** We construct our GKS model on top of an existing KS-like model and add a parameterized correction term $E_\delta$ to it. To be more specific, we define our energy functional to be

$$G[\{\varphi_i\}|\omega] = G_{\text{base}}[\{\varphi_i\}] + E_\delta[\{\varphi_i\}|\omega] \tag{18}$$

where $\omega$ stands for the set of parameters we use in the representation of $E_\delta$. The corresponding single-particle Hamiltonian is then given by

$$\mathcal{H}[\{\varphi_i\}|\omega] = O_{\text{base}}[\{\varphi_j\}] + \mathcal{V}_{\text{ext}} + \mathcal{V}_\delta[\{\varphi_i\}|\omega] \tag{19}$$

The reference point $G_{\text{base}}$ should be a reasonable electron energy functional in KS-like theories, e.g., $G_{\text{HF}}$, $G_{\text{KS; PBE}}$, $G_{\text{Hyb; SCAN0}}$, etc.

Before proceeding further, we list the set of requirements that we ideally want $E_\delta[\{\varphi_i\}|\omega]$ to obey: (1) Generality. The model should be general enough to be applicable for all the systems whose local electronic configurations are well represented by the training data. (2) Locality. The model should be relatively local so that it can potentially be constructed using data from small systems and then be generalizable to larger ones. (3) Symmetry. The model should respect both physical and gauge symmetries. Here, physical symmetry means that $E_c$ should be invariant under translation and rotation of the system. Gauge symmetry means that $E_\delta$ should be invariant when the occupied orbitals $\{|\varphi_i\rangle\}$ undergo a unitary transformation. (4) Accuracy. For target systems, the model should achieve chemical accuracy, i.e., a prediction error lower than 1 kcal/mol. (5) Efficiency. The cost for solving the model should be comparable to that of HF or other DFT models.

To satisfy these requirements, we follow our previous work[8] to construct $E_\delta$ as a neural network model using the "local density matrix" as an input. Briefly speaking, we build our functional based on the one-particle reduced density matrix

$$\Gamma(x, x') = \sum_i \langle x|\varphi_i\rangle\langle\varphi_i|x'\rangle = \sum_i \varphi_i^*(x')\varphi_i(x) \tag{20}$$

We then project it onto a set of atomic basis $\{\alpha^I_{nlm}\}$ indexed by the radial number $n$, azimuthal number $l$, and magnetic (angular) number $m$ and centered on each atom $I$ to get the local density matrix

$$(\mathcal{D}^I_{nl})_{mm'} = \sum_i \langle\alpha^I_{nlm}|\varphi_i\rangle\langle\varphi_i|\alpha^I_{nlm'}\rangle \tag{21}$$

Note here that for simplicity and locality, we only take the block diagonal part of the full matrix; i.e., indices $I$, $n$, and $l$ are taken to be the same for both sides of the projection, and only angular indices $m$ and $m'$ differ. For fast overlap evaluation, we use standard GTO functions but with customized coefficients to make the basis set complete enough. A total of 108 basis functions are used for each atom. The detailed coefficients can be found in the appendix of ref 8.

To deal with the rotational symmetry of the basis $\alpha^I_{nlm}$, we use the eigenvalues of the local density matrix as our descriptor

$$\mathbf{d}^I_{nl} = \text{EigenVals}_{mm'}[(\mathcal{D}^I_{nl})_{mm'}] \tag{22}$$

and we use a neural network model to output the "correction" energy

$$E_\delta = \sum_I \mathcal{F}^{\text{NN}}(\mathbf{d}^I) \tag{23}$$

Hence, the corresponding potential $\mathcal{V}_\delta$ is given by

$$\mathcal{V}_\delta = \sum_{Inlmm'} \frac{\partial E_\delta}{\partial(\mathcal{D}^I_{nl})_{mm'}}|\alpha^I_{nlm}\rangle\langle\alpha^I_{nlm'}| \tag{24}$$



https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

We emphasize that although $E_\delta$ is constructed from the one-particle density matrix, neither the ground-state orbitals nor the density matrix calculated by our model should be expected to have a physical meaning. Instead, we consider the ground-state density to be physical, just as in the standard KS theory, and expect it to coincide with the true ground-state density once we have the exact functional. This is because we follow the GKS approach, rather than an one-electron reduced density matrix functional theory,[17] which cannot be mapped to a KS system.

**2.3. Training Algorithms.** We now discuss how to train a self-consistent model. Here, self-consistency means that the property predicted by the model is obtained via a minimization process and is given at the minimum. A KS-like DFT method is naturally self-consistent. On the contrary, methods like Møller–Plesset perturbation theory[18] and many other post-HF theories are not self-consistent since they do not involve a minimizing procedure. We call those methods energy models to be distinguished from the self-consistent ones. In this context, recent machine learning-based schemes, such as DeePHF method[19] and MOB-ML method,[19] are energy models.

Similar to other supervised learning procedures, we fit the energy functional using existing datasets with certain labels. These labels can be acquired from calculations of high-accuracy methods, such as CCSD(T) and quantum Monte Carlo. Generally speaking, we consider three types of labels:

1. Quantity that is the direct output of the functional after a minimization procedure. Here, it is the total energy.
2. Quantity that depends on both the direct output of the functional and its minimizer. Here, we consider the atomic force.
3. Quantity that depends on the minimizer of the functional but only implicitly through the mathematical form of the functional. Here, we consider the ground-state density.

As has been mentioned, using all these labels in training is a nontrivial task since there is a highly complicated and expensive procedure for calculating the corresponding quantities. Here, we develop general and efficient training algorithms for these three types of labels.

*2.3.1. Type One (Energy).* The training procedure with the energy label may seem straightforward at first glance. Using the $L^2$ norm as the error metric, the optimization problem becomes

$$\min_{\omega} \mathop{\mathbb{E}}_{\text{data}} [(E_{\text{label}} - \min_{\{\varphi_i\},\langle\varphi_i|\varphi_j\rangle=\delta_{ij}} E_{\text{model}}[\{\varphi_i\}|\omega])^2] \tag{25}$$

Here

$$E_{\text{model}}[\{\varphi_i\}|\omega] = G_{\text{base}}[\{\varphi_i\}] + E_{\text{ext}}[\rho[\{\varphi_i\}]] + E_\delta[\{\varphi_i\}|\omega] \tag{26}$$

where the expectation is taken over the training samples.

The gradient of $E_{\text{model}}$ with respect to $\omega$ can be easily obtained using the Hellmann–Feynman theorem. However, the minimization procedure of $\{\varphi_i\}$ involves solving an SCF equation (eq 15) that is very time-consuming. A typical training procedure consists of as many as a million gradient descent steps. This is unrealistic if the SCF equation is solved at every step.

We use a different optimization formalism. Instead of treating the minimized energy as a function of the parameters $\omega$, we consider it as a function of both orbitals $\{\varphi_i\}$ and parameters $\omega$ that satisfies the constraint that $\{\varphi_i\}$ is the minimizer. Therefore, the whole optimization problem can be written as

$$\min_{\omega} \mathop{\mathbb{E}}_{\text{data}} [(E_{\text{label}} - E_{\text{model}}[\{\varphi_i\}|\omega])^2] \tag{27}$$

$$\text{s.t.} \quad \exists\ \varepsilon_i \le \mu,\quad (\mathcal{H}[\{\varphi_i\}|\omega] - \varepsilon_i)|\varphi_i\rangle = 0,\quad \langle\varphi_i|\varphi_j\rangle = \delta_{ij}$$
$$\text{for} \quad i, j = 1...N \tag{28}$$

where eq 28 is a parameterized version of eq 15; i.e., the single-particle Hamiltonian $\mathcal{H}[\{\varphi_i\}|\omega]$ depends on both the orbitals $\{\varphi_i\}$ and the model parameters $\omega$. Here, $\mu$ is the chemical potential and $\varepsilon_1 \le \varepsilon_2 \le ... \le \varepsilon_N$ denote the lowest $N$ eigenvalues.

We now can use a projection method to relax the constraint and this reduces the cost of calculating the SCF equation. In other words, we can first optimize the parameters $\omega$ using an unconstrained gradient-based method with the orbitals $\{\varphi_i\}$ fixed. After several steps, we project the orbitals back to the constraint manifold by solving the SCF equation. Decreasing the projection frequency can largely reduce the computational cost since most of the computation time is spent in the SCF equation. To make it more clear, we write the procedure into the following steps.

1. Initialize a set of $\{\varphi_i\}$ and $\omega$ that satisfies the SCF equation; e.g., take $\omega$ to be all zero and $\{\varphi_i\}$ to be the Hartree–Fock solution. Also, keep track of the predicted energy $E_{\text{model}}$.
2. Update the parameters $\omega$ by training the model following eq 27 with fixed orbitals $\{\varphi_i\}$.
3. Update the orbitals $\{\varphi_i\}$ by solving the SCF equation with fixed model parameters $\omega$.
4. Check whether the predicted energy $E_{\text{model}}$ converges. If not, go to step 2 and do more iterations.

A schematic illustration of this approach is shown in Figure 1. Note that we usually take many training steps in step 2. In practice, when restarting from old parameters using new orbitals,

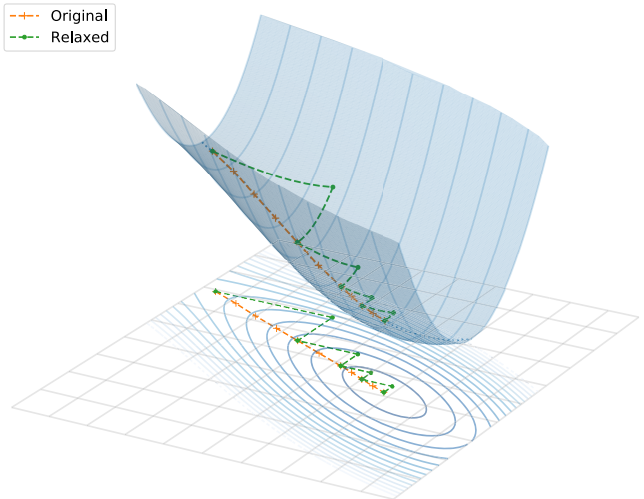



**Figure 1.** Schematic illustration of the iterative training procedure using the projection method. Here, "Original" stands for the direct constrained minimization following eq 25 and "Relaxed" stands for the relaxed projection method in eqs 27 and 28. In step 2 of the relaxed method, the optimization of $\omega$ strays away from the minimizing manifold, while in step 3, the projection of $\{\varphi_i\}$ brings it back.



https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

we find it possible to train the model until the validation error no longer decreases without breaking the convergence of the whole procedure. Therefore, the total time of solving the SCF equation is significantly reduced.

We note that a similar formalism has been proposed and used by the NeuralXC scheme.[11] The major difference is that in DeePKS, a single NN function is used as a universal approximator. The function form does not change with the iterative process, and its parameters do not depend on the chemical species of the associated atom. In contrast, in NeuralXC, the parameters depend on the chemical species, and in each iteration, a new NN layer is appended to the NN model from the previous iteration. The reformulation in DeePKS is designed to make it more transferable to larger chemical space and more suited for larger dataset.

*2.3.2. Type Two (Force).* The atomic forces from the proposed model can be easily calculated by the standard Hellmann−Feynman theorem

$$\begin{aligned} &F_{\text{model}}[\{\varphi_i^*[\omega]\}|\omega] \\ &= -\frac{\partial E_{\text{model}}[\{\varphi_i^*\}|\omega]}{\partial X} \\ &= F_{\text{base}}[\{\varphi_i^*\}] \\ &\quad - \sum_{Inlmm'} \frac{\partial E_\delta[\{\varphi_i^*\}|\omega]}{\partial (\mathcal{D}^I_{nl})_{mm'}} \sum_i \left\langle \varphi_i^* \middle| \frac{\partial (|\alpha^I_{nlm}\rangle\langle\alpha^I_{nlm}|)}{\partial X} \middle| \varphi_i^* \right\rangle \end{aligned} \tag{29}$$

where we have written out the dependence on the parameters $\omega$ explicitly. We use $\{\varphi_i^*\}$ to denote the minimizer of the total energy functional, which themselves are functions of $\omega$

$$\{\varphi_i^*[\omega]\} = \arg\min_{\langle\varphi_i|\varphi_j\rangle=\delta_{ij}} E_{\text{model}}[\{\varphi_i\}|\omega] \tag{30}$$

We can see that the force $F$ depends directly on both the model parameters $\omega$ and the minimizing orbitals $\{\varphi_i^*[\omega]\}$. This introduces an additional difficulty when we evaluate the gradient of $F$ with respect to $\omega$. The contribution from the $\{\varphi_i^*[\omega]\}$ term is very hard to compute since it involves a whole minimization procedure, and there is no Hellmann−Feynman theorem to save us.

Luckily, this difficulty disappears in our iterative training procedure, where the gradient we used to optimize $\omega$ is no longer the constrained one. The orbitals are treated as independent variables so that they do not contribute to the gradient. Therefore, the gradient can be calculated straightforwardly using a back-propagation procedure.

By writing the force term into the loss function, the new optimization problem becomes

$$\begin{aligned} \min_{\omega} \quad & \mathbb{E}_{\text{data}}[(E_{\text{label}} - E_{\text{model}}[\{\varphi_i\}|\omega]) \\ & \quad + \lambda_f (F_{\text{label}} - F_{\text{model}}[\{\varphi_i\}|\omega])^2] \\ \text{s. t.} \quad & \exists\ \varepsilon_i \le \mu, \\ & (\mathcal{H}[\{\varphi_i\}|\omega] - \varepsilon_i)|\varphi_i\rangle = 0, \\ & \langle\varphi_i|\varphi_j\rangle = \delta_{ij} \\ \text{for} \quad & i, j = 1 \dots N \end{aligned} \tag{31}$$

where $\lambda_f$ is a tunable parameter that determines the weight of force label in the loss function. We can then use the iterative algorithms described above to solve this optimization problem.

*2.3.3. Type Three (Density).* The ground-state density given by the proposed model is a function of the minimizing orbitals

$$\rho_{\text{model}}(x) = \sum_i^N |\varphi_i^*(x)|^2 \tag{32}$$

Since it does not depend on the parameters $\omega$ explicitly, unlike the case for forces, we cannot write the density into the loss function. To solve this problem, we introduce a penalty term in the SCF equation to "guide" the training procedure. This is done by changing the minimization problem in eq 13 into

$$\min_{\{\varphi_i\},\langle\varphi_i\varphi_j\rangle=\delta_{ij}} \{E_{\text{model}}[\{\varphi_i\}|\omega] + \lambda_\rho D[\rho[\{\varphi_i\}], \rho_{\text{label}}]\} \tag{33}$$

where $\lambda_\rho > 0$ is the strength of the penalty and $D$ is some non-negative error metric that equals to zero only when $\rho = \rho_{\text{label}}$. Hence, if the SCF solution gives the exact density, then the penalty term does not influence the minimizer. Otherwise, according to the discussion in Appendix A, because of an additional potential term in the SCF equation

$$\mathcal{V}_{\text{pnt}}[\rho|\rho_{\text{label}}] = \frac{\delta D[\rho, \rho_{\text{label}}]}{\delta\rho} \tag{34}$$

it will lead to a self-consistent energy strictly larger than the one obtained without this penalty term and a density that is closer to the label.

Note here that $\lambda_\rho$ does not need to be a fixed value. Rather, it can be a bunch of values or even a non-negative random variable. When the model yields exact density, all the functionals with different $\lambda_\rho$ values should give the exact solution. When the solution is not exact, randomized $\lambda_\rho$ serves as a regulator that helps reduce the overfitting and provides better results compared to using a single fixed value and is more efficient than using multiple values. If we choose it to be a random variable, then the modified optimization problem becomes

$$\begin{aligned} \min_{\omega} \quad & \mathbb{E}_{\text{data},\lambda_\rho}[(E_{\text{label}} - E_{\text{model}}[\{\varphi_i\}|\omega])^2 \\ & \quad + \lambda_f (F_{\text{label}} - F_{\text{model}}[\{\varphi_i\}|\omega])^2] \\ \text{s. t.} \quad & \exists\ \varepsilon_i \le \mu, \\ & (\mathcal{H}[\{\varphi_i\}|\omega] + \lambda_\rho \mathcal{V}_{\text{pnt}}[\rho[\{\varphi_i\}]|\rho_{\text{label}}] - \varepsilon_i)|\varphi_i\rangle = 0, \\ & \langle\varphi_i|\varphi_j\rangle = \delta_{ij} \\ \text{for} \quad & i, j = 1 \dots N \end{aligned} \tag{35}$$

The same projection-based training procedure can be applied to this loss function.

We note that although in this paper, we take energy, force, and density as examples, these algorithms are rather general and can be easily transferred to similar learning problems that involve an optimization procedure for the evaluation of meaningful quantities. For example, if we include dipole as a label, then we can add a penalty term similar to eq 33. Moreover, the training algorithms are not limited to the specific GKS model we described above. Instead, they can be applied to gradient-based optimization tasks for any exchange correlation functionals and even other self-consistent learning problems.

**2.4. Related Works.** Before reporting numerical results, we discuss a few related works to the DeePKS scheme, in the spirit


https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

of developing machine learning-assisted physical models. First, there have been some efforts on using deep neural networks to parameterize the many-electron-ion trial wave function and using a variational Monte Carlo (VMC) approach to optimize the parameters. The first attempt was reported by ref 20. This is followed by some more recent efforts.[21,22] The purpose of these efforts is to solve the original quantum many-body electron problem. In comparison, DeePKS takes results from the quantum many-body electron problem as inputs and attempts to parameterize the exchange-correlation functionals.

Second, there have been some efforts on using machine learning-based schemes to represent quantities that are functions of atomic positions and their chemical species. An incomplete list includes refs 23−36. In particular, ref 25 reports a kernel-based method for fast and accurate modeling of molecular atomization energies; ref 26 reports a Δ-learning approach, which shares a similar spirit of our work in a different context. Such an idea of fitting the difference between a baseline model and target values has been widely adopted by the machine learning community, see, for example, the gradient boosting machine[37] that iterates the delta fitting procedure for multiple times in a more systematic way.

## 3. RESULTS

We now examine the performance of the DeePKS scheme on three classes of data that have been used for benchmark purposes in the literature. Unless otherwise specified, all labels are given by CCSD(T), and all calculations are conducted using the cc-pVDZ basis.

- Malonaldehyde including 1500 configurations with energy, force, and density labels. We use this dataset to test thoroughly our training method with all three types of labels. Since, within the CCSD(T) formalism, perturbative triple does not give the corresponding density, we use CCSD for density-related tests. The data are calculated from PySCF[38] with molecular configurations coming from the sGDML dataset.[39]
- Three molecules (malonaldehyde, benzene, and toluene) including 1500 configurations for each molecule with energy and force labels. This is a subset of the sGDML dataset[39] under the same numerical setup; therefore, we can train one model on all three molecules and examine the intermolecule performance as a first step toward universal functionals.
- QM7b-T dataset[40] including 7212 molecules and one configuration for each molecule, with energy labels only. This is the largest publicly available dataset with CCSD(T) accuracy. It has been used to benchmark several other methods.[19,41,42] as well as the energy model developed in ref 8. We test it here to make a comparison of the new self-consistent model and previous energy model. We also use it to examine the ability of the DeePKS method for generating "universal" functionals that are applicable to as many systems as possible.

We emphasize that the objective of our method is to build one single functional with chemical accuracy for as many systems as possible, although it is currently limited by the data we have. The functional should be able to predict accurate results for all the systems that are well represented in the training set, and its coverage can be enlarged continuously by adding more and more training data.

We implement the DeePKS method using the open-source packages PySCF[38] and PyTorch.[43] We start our iteration from a functional obtained using DeePHF and orbitals solved from that functional. In each iteration, the optimization of the neural network parameters is conducted for 10,000 epochs using the ADAM optimizer.[44] For all training that includes force labels, we set the parameter $\lambda_f$ to be 0.1. One more trick we use is that to speed up the convergence, after training with ADAM, we further correct the model with a global energy shift, which is calibrated from the training set.

We now examine the performance of our method on the malonaldehyde molecule using the HF functional as the base model, $G_{base} = G_{HF}$. As a first step, to have an intuitive picture of the newly proposed iterative method, we use energy and force as training labels and study the behavior of the mean absolute error (MAE) in the testing set during the training process. The error for the forces is calculated component-wise. The training is done on 1000 molecular configurations and testing on the remaining 500. We also include results from sGDML[39] and DeePMD models[31,32] for comparison.

As can be seen in Figure 2, when training with only energy labels (iterations 0 to 6), the testing accuracy quickly saturates,

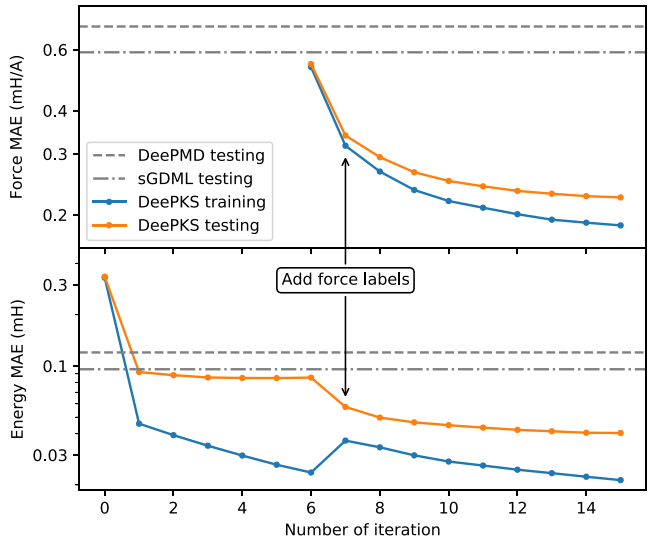



**Figure 2.** Energy and force errors during the training process for the malonaldehyde dataset. Force labels are added starting from iteration 7. Results from DeePMD and sGDML are included for comparison.

while the training error keeps decreasing, suggesting that the model begins to overfit. On the other hand, even though we train with only energy labels, the model already outperforms DeePMD and sGDML methods, both of which utilize forces as training labels. When we include force labels after iteration 6, the testing accuracy can be further improved two to three times. This shows the effectiveness of adding force labels in the training.

To further examine the sample efficiency of our method, we study the learning curve associated with the malonaldehyde molecule by plotting the testing MAE of both energies and forces versus the number of training samples. Each time the dataset is augmented, existing samples in the dataset are kept, and the testing error is calculated on the remaining part of the data. For comparison, we include the results of NeuralXC[11] and DeePMD. As shown in Figure 3, in all cases, DeePKS outperforms both DeePMD and NeuralXC: Using the same


https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

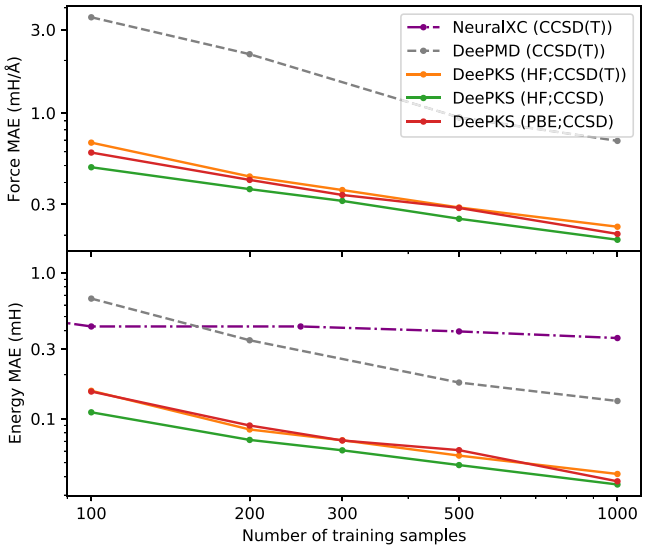



**Figure 3.** Learning curve of both energy and force for the malonaldehyde dataset. Results from DeePMD and NeuralXC are included for comparison. NeuralXC results are digitally captured from ref 11.

amount of training data, the accuracy of both the energies and forces is improved 3 to 10 times. As an ablation study, we also examine the situation of using labels at the CCSD level and, starting from PBE functionals[6] ($G_{\text{base}} = G_{\text{KS; PBE}}$), we find that the results do not change much. Therefore, hereafter, we focus on the HF-based model since the implementation of PBE in PySCF is rather slow.

We now move to density-related tests. Here, we use labels at the CCSD level. We follow eq 35 to train our model with density labels. The error penalty term is taken to be the Coulomb repulsion energy of the density difference

$$D[\rho, \rho_{\text{label}}] = \int \mathrm{d}x_1\, \mathrm{d}x_2 \frac{\Delta\rho(x_1)\Delta\rho(x_2)}{|x_1 - x_2|}$$
$$\Delta\rho(x) = \rho(x) - \rho_{\text{label}}(x) \tag{36}$$

which can be evaluated with very small cost in PySCF. The penalty parameter $\lambda_\rho$ is sampled uniformly from 0 to 1 for every data point and every SCF calculation. We train with this setup for 20 iterations and then remove the penalty and perform another 5 iterations for relaxation. As we will see later, such relaxation will slightly reduce the accuracy for density but substantially improve the accuracy for energy and force.

We study the performance of the DeePKS model in terms of the prediction error of energy $E$, force $F$, dipole $\mu$, and point-wise electron density $\rho$. For comparison, we also include different training schemes and results from several other methods. We use the $l^1$ norm for energy and density, the component-wise $l^1$ norm for force, and $l^2$ norm for dipole as error metrics.

All models are trained on 1000 malonaldehyde configurations and the testing errors are averaged over the remaining 500 configurations. For HF and DFT functionals, a constant energy shift, calculated from the training set, is applied to their predicted total energy. Our testing results are summarized in Table 1.

In general, we find ML-based methods perform much better than traditional HF or DFT functionals in terms of the accuracy of energy and forces. This is expected since these methods are directly trained with corresponding labels on this specific system. Traditional functionals, on the other hand, give rather good prediction on dipoles and densities. Only by including density labels can DeePKS outperform the state-of-the-art conventional functional (SCAN0). It is also interesting to observe that even without dipole labels, the DeePKS models, obtained in different ways, significantly outperform HF, PBE, and SCAN0 in terms of testing accuracy on dipole moments.

**Table 1. Comparison of Different Methods in Terms of the Prediction Errors for Energy, Force, Dipole, and Density for the Malonaldehyde Dataset**[a]

| method | | $\|\Delta E\|_1$ | $\|\Delta F\|_1$ | $\|\Delta\mu\|_2$ | $\|\Delta\rho\|_1$ |
|---|---|---|---|---|---|
| sGDML | (w/ $E$, $F$) | 0.10 | 0.59 | – | – |
| DeePMD | (w/ $E$, $F$) | 0.13 | 0.69 | – | – |
| HF | ($E$ + 805.19) | 3.29 | 24.1 | 0.66 | 0.58 |
| PBE | ($E$ − 400.33) | 1.35 | 7.53 | 0.17 | 0.35 |
| SCAN0 | ($E$ − 522.05) | 1.83 | 10.9 | 0.32 | 0.29 |
| DeePKS | (w/ $E$) | 0.067 | 0.44 | 0.10 | 0.50 |
| DeePKS | (w/ $E$, $F$) | 0.034 | 0.18 | 0.10 | 0.39 |
| DeePKS | (w/ $E$, $F$, $\rho$) | 0.048 | 0.30 | 0.044 | 0.20 |
| DeePKS | (w/ $E$, $F$, $\rho$; rlxd) | 0.041 | 0.24 | 0.047 | 0.21 |

[a]Errors are measured in mH for energy, mH/Å for force, Debye for dipole, and e for density. For ML-based methods (sGDML, DeePMD, and DeePKS), the types of labels used for training are shown in parentheses. The term "rlxd" stands for the relaxation procedure after training with density.

For a more intuitive view, we compare the ground-state density given by SCAN0 with different training schemes for DeePKS. We take a sliced line that crosses an oxygen atom, and we plot the density difference compared with the CCSD label. As shown in Figure 4, when training without density, the error is

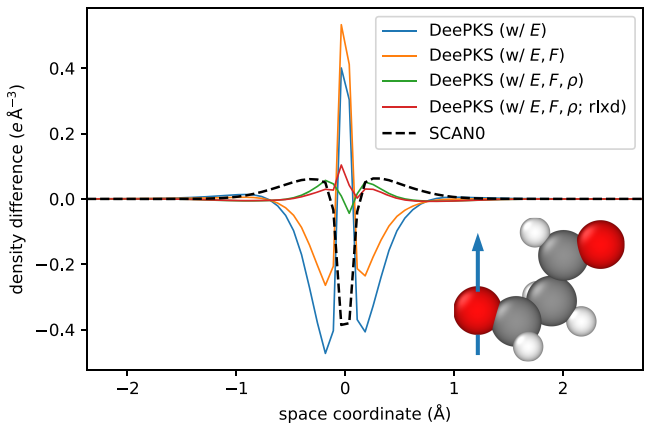



**Figure 4.** Density difference with respect to $\rho_{\text{CCSD}}$ on a sliced line crossing an oxygen atom in the malonaldehyde molecule, given by different training schemes for DeePKS and the SCAN0 functional. The $x$ axis corresponds to space coordinates on the sliced line, which is shown in the inset as a blue arrow. The molecule is drawn by OVITO.[45]

relatively large (around 0.5 e Å$^{-3}$ at maximum) in the core region and is worse than the SCAN0 prediction. After we add density labels, the error is reduced to below 0.1 e Å$^{-3}$, showing the necessity of using density labels in the training. We also note that the absolute density value can reach 600 e Å$^{-3}$ at the core; hence, even the largest difference in density is still very small compared to the absolute value.


https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

As a further step, we test the performance of DeePKS on learning one single functional for multiple molecules simultaneously. This is in general a hard task, especially when the number of training samples is very limited. As mentioned in ref 39, for these so-called transferable models, "energy prediction errors are often much larger than 1 kcal/mol", even with a huge amount of training data.[29,46,47] However, this step is crucial and inevitable since our ultimate goal is to build one universally accurate functional for a wide range of systems.

We then check the behavior of DeePKS for fitting malonaldehyde, benzene, and toluene at the same time, with energy and force labels. This is the largest set of data we find with both energy and force at the CCSD(T) level calculated in the same numerical setup. We take 1000 samples for each molecule in the training and test on the remaining configurations. We summarize our results in Table 2, including a comparison with

**Table 2. Comparison of Different Methods on the Prediction Accuracy of Energy and Force for the Dataset Containing Configurations of Malonaldehyde, Benzene, and Toluene[a]**

| | malonaldehyde | | benzene | | toluene | |
|---|---|---|---|---|---|---|
| method | $\|\|\Delta E\|\|_1$ | $\|\|\Delta F\|\|_1$ | $\|\|\Delta E\|\|_1$ | $\|\|\Delta F\|\|_1$ | $\|\|\Delta E\|\|_1$ | $\|\|\Delta F\|\|_1$ |
| NeuralXC* | 0.35 | – | 0.075 | – | 0.20 | – |
| sGDML* | 0.10 | 0.59 | 0.006 | 0.06 | 0.05 | 0.33 |
| DeePKS* | 0.04 | 0.22 | 0.007 | 0.07 | 0.06 | 0.32 |
| DeePKS | 0.07 | 0.41 | 0.014 | 0.13 | 0.08 | 0.42 |

[a]Errors are measured in mH for energy and mH/Å for force. Force errors are calculated component-wise. Methods marked with "*" are trained separately on each molecule. NeuralXC results are digitally captured from ref 11.

NeuralXC and sGDML. Despite a small loss in accuracy, the DeePKS method is still comparable with sGDML and outperforms NeuralXC, both of which are trained separately on each individual molecule. We also note that sGDML performs relatively well on benzene and toluene, possibly due to their explicit handling of the point group symmetry. Such treatment can improve the sample efficiency for highly symmetric molecules like benzene and toluene yet may not be very helpful for more general molecules.

For a larger test, we examine the performance of DeePKS on the QM7b-T dataset. This is the largest dataset we have with CCSD(T) level of energy and is also used for benchmarking the energy model DeePHF.[8] We study the learning curve by randomly selecting some samples as a training set and test on the rest. Since there is no new label included and the model is trained only with energy, we should not expect DeePKS to exhibit any accuracy improvement with respect to DeePHF. The best result we can look for is that the self-consistent model behaves as well as the energy model. This is indeed the case, as shown in Figure 5.

We further examine the transferability of DeePKS to much larger systems by predicting hydrocarbon reaction and isomerization energies using the HC7[48] and ISOL6[49] benchmarks. The 7000 samples randomly selected from the QM7b-T dataset, used to train the DeePKS model, contain at most 7 heavy atoms. However, HC7 and ISOL6 contain at most 12 and 15 heavy atoms, respectively. As shown in Table 3, DeePKS outperforms conventional DFT functionals and generalizes better than the current best-performing ML-based model ANI1-ccx, which is trained using a huge dataset of 5M molecular configurations with

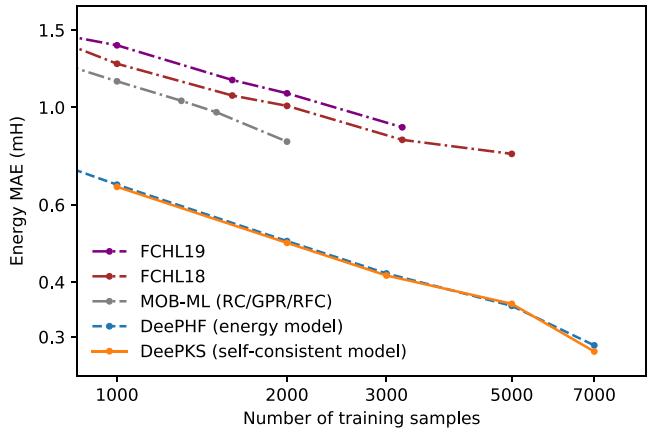



**Figure 5.** Learning curve of DeePHF and DeePKS methods on the QM7b-T dataset. Results of MOB-ML with regression clustering[41] and FCHL[42] methods are included for comparison. The FCHL results use MP2 energy as training and testing labels and are digitally captured from ref 42.

**Table 3. MAE of Reaction and Isomerization Energies Calculated Using the HC7[48] and ISOL6[49] Datasets, Respectively[a]**

| method | HC7 | ISOL6 |
|---|---|---|
| PBE | 8.91 | 3.80 |
| SCAN | 16.22 | 3.25 |
| B3LYP | 16.74 | 4.16 |
| SCAN0 | 23.94 | 3.65 |
| $\omega$B97X | 17.71 | 3.45 |
| $\omega$B97M-V | 5.86 | 3.81 |
| | | |
| ANI-1ccx | 3.24 | 2.41 |
| DeePKS | 2.88 | 1.26 |

[a]Results of ANI-1ccx are taken from ref 47. Errors are given in mH. All DFT methods' results are obtained using the cc-pVDZ basis. The MAEs of DFT methods (including DeePKS) are calculated by comparing with results from CCSD(T)/cc-pVDZ calculation. The MAE of ANI-1ccx is calculated by comparing with the methods used for generating their training data, i.e., CCSD(T)*/CBS.

DFT energies and forces and fine-tuned on about 500K configurations with CCSD(T)*/CBS energies.

As a final remark, we show that the DeePKS model can indeed be evaluated efficiently. Figure 6 shows the computational cost of different methods for calculating alkanes ranging from one to seven carbon atoms. The number of iterations in all SCF-based methods is set to 10. We note that for PBE and other DFT functionals, the implementation in PySCF involves numerical integration over space grids, which is much more expensive for small molecules with the GTO basis set, wherein analytical evaluations of orbital overlapping can be carried out efficiently in the HF method and the HF-based DeePKS. As a result, DeePKS is even faster than PBE and scales similarly with HF. The additional cost over HF scales essentially linearly with respect to system size. For larger systems where the $O(N^4)$ scaling in HF begins to dominate, we can switch to PBE or other KS functionals as the starting point and implement our method in a planewave framework to retain the cubic scaling. The planewave implementation of DeePKS is left for future work.


https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

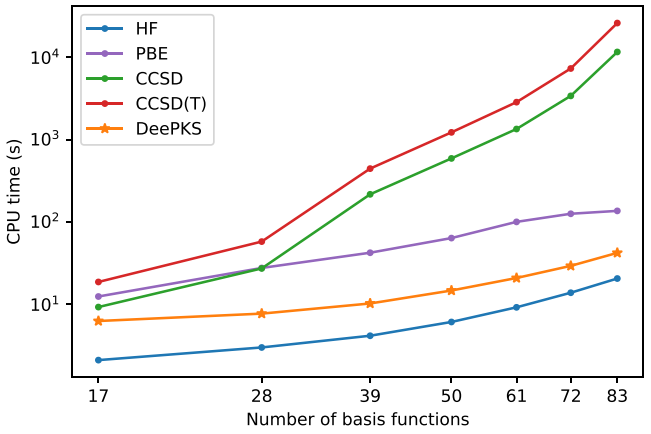



**Figure 6.** CPU time spent in the calculations of alkanes using different methods.

## 4. CONCLUSIONS

We presented a general framework for learning chemically accurate self-consistent energy functionals using different types of labels, including energy, force, and density. The new training method, combined with a self-consistent extension of DeePHF, leads to a generalized Kohn−Sham functional with the accuracy of CCSD(T) and the computational cost of DFT. We examined the performance of the proposed method on multiple molecular datasets and obtained highly accurate predictions for multiple properties like energy, force, and density. In addition, the proposed method is capable of learning a single functional that covers different molecular systems, and its accuracy can be continuously improved by adding more training data. We believe that it is a good starting point toward a universally accurate functional for molecules, and we are confident that it can be extended to include condensed phases.

## APPENDIX A

### A.1. Properties of the Modified Minimization Scheme for Density Optimization

We discuss the properties of the energy and density when we modify in eq 33 the minimization scheme for density optimization. For simplicity, let us use the following notation

$$L^{\lambda_\rho}[\{\varphi_i\}] = E_{\text{model}}[\{\varphi_i\}] + \lambda_\rho D[\rho[\{\varphi_i\}], \rho_{\text{label}}] \tag{37}$$

and

$$\{\varphi_i^{\lambda_\rho}\} = \arg\min_{\langle\varphi_i|\varphi_j\rangle=\delta_{ij}} L^{\lambda_\rho}[\{\varphi_i\}] \tag{38}$$

for which we assume that the global minimizer of $L^{\lambda_\rho}[\{\varphi_i\}]$ is unique. In particular

$$\{\varphi_i^*\} = \arg\min_{\langle\varphi_i|\varphi_j\rangle=\delta_{ij}} L^0[\{\varphi_i\}] = \arg\min_{\langle\varphi_i|\varphi_j\rangle=\delta_{ij}} E_{\text{model}}[\{\varphi_i\}] \tag{39}$$

gives the original minimizer.

For $\lambda_1 > \lambda_2 \geq 0$, we have

$$L^{\lambda_1}[\{\varphi_i^{\lambda_2}\}] \geq L^{\lambda_1}[\{\varphi_i^{\lambda_1}\}] \tag{40}$$

$$L^{\lambda_1}[\{\varphi_i^{\lambda_1}\}] \geq L^{\lambda_2}[\{\varphi_i^{\lambda_1}\}] \tag{41}$$

$$L^{\lambda_2}[\{\varphi_i^{\lambda_1}\}] \geq L^{\lambda_2}[\{\varphi_i^{\lambda_2}\}] \tag{42}$$

Equation 40 holds since $\{\varphi_i^{\lambda_1}\}$ is the minimizer of $L^{\lambda_1}$. Similarly, eq 42 holds since $\{\varphi_i^{\lambda_2}\}$ is the minimizer of $L^{\lambda_2}$; eq 41 holds since the term $(\lambda_1 - \lambda_2)D[\rho[\{\varphi_i^{\lambda_1}\}], \rho_{\text{label}}]$ is non-negative.

It is straightforward to see that equalities hold for all these equations if and only if both $D[\rho[\{\varphi_i^{\lambda_1}\}], \rho_{\text{label}}]$ and $D[\rho[\{\varphi_i^{\lambda_2}\}], \rho_{\text{label}}]$ are 0. In this case, both the energy $L^{\lambda_\rho}[\{\varphi_i^{\lambda_\rho}\}]$ and the minimizing density $\rho[\{\varphi_i^{\lambda_\rho}\}]$ will be the same for all $\lambda_\rho \geq 0$. Otherwise, we will have the following two properties:

1. $E_{\text{model}}[\{\varphi_i^{\lambda_\rho}\}]$ is strictly larger than $E_{\text{model}}[\{\varphi_i^*\}]$ by taking $\lambda_1 = \lambda_\rho$ and $\lambda_2 = 0$ in eq 42.
2. A larger penalty will lead to a density that is closer to the label. This can be obtained by adding eq 40 to eq 42, which will lead to

$$(\lambda_1 - \lambda_2)(D[\rho[\{\varphi_i^{\lambda_2}\}], \rho_{\text{label}}] - D[\rho[\{\varphi_i^{\lambda_1}\}], \rho_{\text{label}}]) \geq 0 \tag{43}$$

Therefore, we have $D[\rho[\{\varphi_i^{\lambda_1}\}], \rho_{\text{label}}] \leq D[\rho[\{\varphi_i^{\lambda_2}\}], \rho_{\text{label}}]$.

### A.2. Visualization of Molecular Orbitals of Malonaldehyde

For an intuitive picture on how DeePKS works, we provide in Figure A1 a comparison plot for the highest occupied molecular orbital (HOMO) of malonaldehyde molecule, calculated by four different models, including HF theory, KSDFT with PBE functional, DeePKS based on HF, and DeePKS based on PBE. It can be seen that the two DeePKS models behave similarly. The difference between the two DeePKS models is much smaller than that between the methods they are based on, i.e., HF and PBE. This is well expected since DeePKS approximates the "exact" functional that gives the same prediction of its labeling method (CCSD in this case) and should be insensitive to its starting point. We note again that the orbitals predicted by DeePKS models have no physical meaning. They are shown here as an indication of the robustness of the DeePKS method.

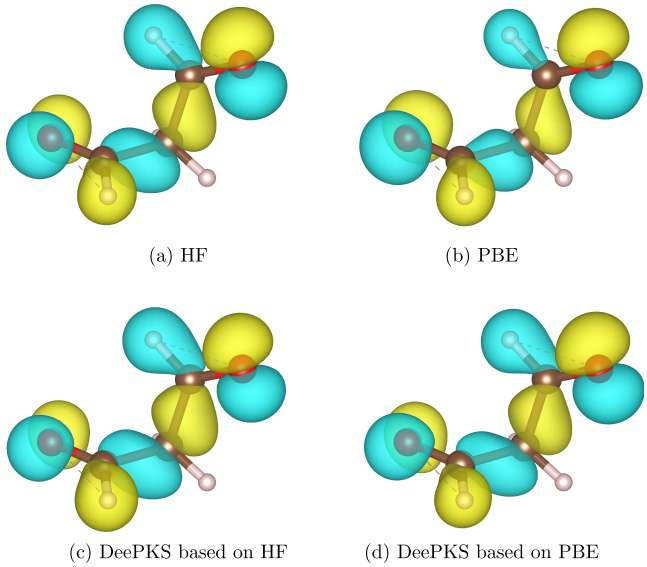



**Figure A1.** (a−d) HOMO of a typical configuration of malonaldehyde molecule calculated by four different models. The isosurfaces are drawn at the level of 0.05 a.u. Visualization is done by VESTA software.[50]

### A.3. Integrated Absolute Density Difference of Malonaldehyde

We show in Figure A2 the integrated absolute difference of density calculated by different models. The difference is plotted against one spatial direction with the other two being integrated.



https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

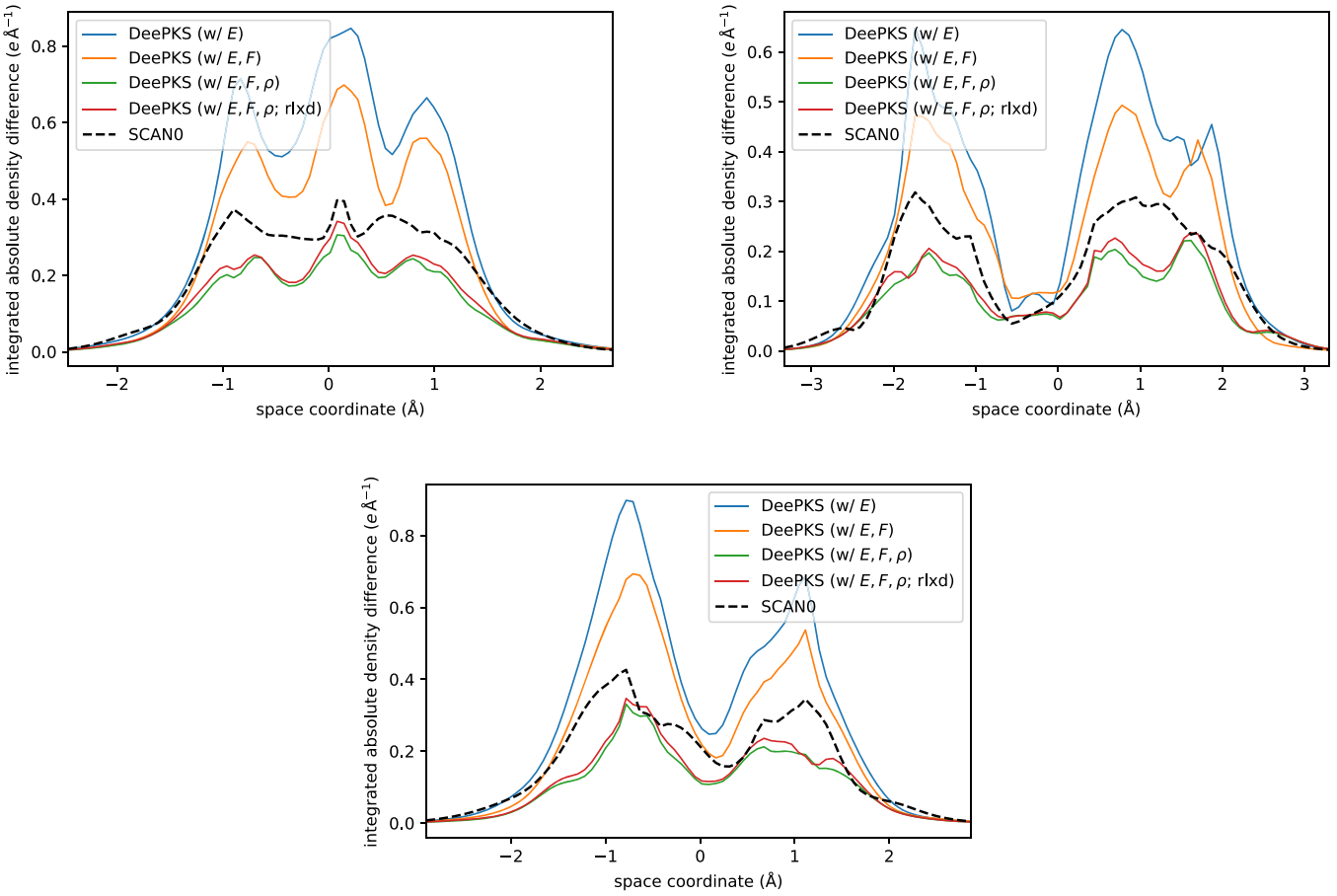



**Figure A2.** Integrated absolute density difference of a typical configuration of the malonaldehyde molecule. The absolute density difference is integrated on two spatial dimensions and plotted against the third dimension. The subfigures correspond to different integrated space dimensions.

The findings are similar to the ones shown in Figure 4. In all cases, the error in density from DeePKS models can be largely reduced by using density as labels in the training procedure. The models trained with density labels can give more accurate density prediction than the SCAN0 functional.

### A.4. Learning Curve of Three Molecules

We provide in Figure A3 the learning curve of DeePKS trained on the dataset containing snapshots of malonaldehyde, benzene, and toluene molecules at the same time.

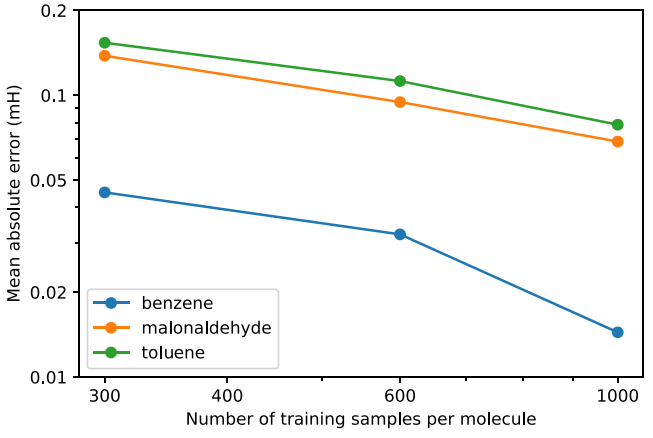



**Figure A3.** Learning curve of the DeePKS method on the dataset containing snapshots of malonaldehyde, benzene, and toluene molecules. For each choice of the training data size, a single DeePKS model is trained for all three molecules simultaneously.

## AUTHOR INFORMATION


### Corresponding Authors

**Linfeng Zhang** − *Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544, United States;* orcid.org/0000-0002-8470-5846; Email: linfengz@princeton.edu

**Han Wang** − *Laboratory of Computational Physics, Institute of Applied Physics and Computational Mathematics, Beijing 100088, People's Republic of China;* orcid.org/0000-0001-5623-1148; Email: wang_han@iapcm.ac.cn

### Authors

**Yixiao Chen** − *Program in Applied and Computational Mathematics, Princeton University, Princeton, New Jersey 08544, United States;* orcid.org/0000-0001-8201-5887

**Weinan E** − *Program in Applied and Computational Mathematics and Department of Mathematics, Princeton University, Princeton, New Jersey 08544, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acs.jctc.0c00872


### Notes

The authors declare no competing financial interest.



https://dx.doi.org/10.1021/acs.jctc.0c00872
J. Chem. Theory Comput. 2021, 17, 170−181

## ■ ACKNOWLEDGMENTS

We thank Xiao Wang and Lin Lin for beneficial discussions. The work of Y.C., L.Z., and W.E. was supported in part by a gift from iFlytek to Princeton University, the ONR grant N00014-13-1-0338, and the Center Chemistry in Solution and at Interfaces (CSI) funded by the DOE Award DE-SC0019394. The work of H.W. is supported by the National Science Foundation of China under grant no. 11871110, the National Key Research and Development Program of China under grant nos. 2016YFB0201200 and 2016YFB0201203, and Beijing Academy of Artificial Intelligence (BAAI).

## ■ REFERENCES

(1) Hohenberg, P.; Kohn, W. Inhomogeneous electron gas. *Phys. Rev.* **1964**, *136*, B864.
(2) Pople, J. A.; Head-Gordon, M.; Raghavachari, K. Quadratic configuration interaction. A general technique for determining electron correlation energies. *J. Chem. Phys.* **1987**, *87*, 5968−5975.
(3) Jeziorski, B.; Monkhorst, H. J. Coupled-cluster method for multideterminantal reference states. *Phys. Rev. A* **1981**, *24*, 1668.
(4) Kohn, W.; Sham, L. J. Self-consistent equations including exchange and correlation effects. *Phys. Rev.* **1965**, *140*, A1133.
(5) Seidl, A.; Görling, A.; Vogl, P.; Majewski, J. A.; Levy, M. Generalized Kohn-Sham schemes and the band-gap problem. *Phys. Rev. B* **1996**, *53*, 3764.
(6) Perdew, J. P.; Burke, K.; Ernzerhof, M. Generalized gradient approximation made simple. *Phys. Rev. Lett.* **1996**, *77*, 3865.
(7) Sun, J.; Ruzsinszky, A.; Perdew, J. P. Strongly Constrained and Appropriately Normed Semilocal Density Functional. *Phys. Rev. Lett.* **2015**, *115*, No. 036402.
(8) Chen, Y.; Zhang, L.; Wang, H.; E, W. Ground State Energy Functional with Hartree−Fock Efficiency and Chemical Accuracy. *J. Phys. Chem. A* **2020**, *124*, 7155−7165.
(9) Snyder, J. C.; Rupp, M.; Hansen, K.; Müller, K.-R.; Burke, K. Finding density functionals with machine learning. *Phys. Rev. Lett.* **2012**, *108*, 253002.
(10) Bogojeski, M.; Vogt-Maranto, L.; Tuckerman, M. E.; Mueller, K.-R.; Burke, K. Density functionals with quantum chemical accuracy: From machine learning to molecular dynamics. *ChemRxiv* **2019**, 8079917.
(11) Dick, S.; Fernandez-Serra, M. Machine learning accurate exchange and correlation functionals of the electronic density. *Nat. Commun.* **2020**, *11*, 3509.
(12) Lei, X.; Medford, A. J. Design and analysis of machine learning exchange-correlation functionals via rotationally invariant convolutional descriptors. *Phys. Rev. Mater.* **2019**, *3*, No. 063801.
(13) Liu, Q.; Wang, J.; Du, P.; Hu, L.; Zheng, X.; Chen, G. Improving the Performance of Long-Range-Corrected Exchange-Correlation Functional with an Embedded Neural Network. *J. Phys. Chem. A* **2017**, *121*, 7273−7281.
(14) Nagai, R.; Akashi, R.; Sugino, O. Completing density functional theory by machine learning hidden messages from molecules. *npj Comput. Mater.* **2020**, *6*, 1−8.
(15) Levy, M. Universal variational functionals of electron densities, first-order density matrices, and natural spin-orbitals and solution of the v-representability problem. *Proc. Natl. Acad. Sci. U. S. A.* **1979**, *76*, 6062−6065.
(16) Lieb, E. H. Density functionals for coulomb systems. *Int. J. Quantum Chem.* **1983**, *24*, 243−277.
(17) Gilbert, T. L. Hohenberg-Kohn theorem for nonlocal external potentials. *Phys. Rev. B* **1975**, *12*, 2111.
(18) Møller, C.; Plesset, M. S. Note on an approximation treatment for many-electron systems. *Phys. Rev.* **1934**, *46*, 618.
(19) Cheng, L.; Welborn, M.; Christensen, A. S.; Miller, T. F., III A universal density matrix functional from molecular orbital-based machine learning: Transferability across organic molecules. *J. Chem. Phys.* **2019**, *150*, 131103.
(20) Han, J.; Zhang, L.; E, W. Solving many-electron Schrödinger equation using deep neural networks. *J. Comput. Phys.* **2019**, *399*, 108929.
(21) Hermann, J.; Schätzle, Z.; Noé, F. Deep-neural-network solution of the electronic Schrödinger equation. *Nat. Chem.* **2020**, 891−897.
(22) Pfau, D.; Spencer, J. S.; Matthews, A. G. D. G.; Foulkes, W. M. C. *Ab initio* solution of the many-electron Schrödinger equation with deep neural networks. *Phys. Rev. Res.* **2020**, *2*, No. 033429.
(23) Behler, J.; Parrinello, M. Generalized neural-network representation of high-dimensional potential-energy surfaces. *Phys. Rev. Lett.* **2007**, *98*, 146401.
(24) Bartók, A. P.; Payne, M. C.; Kondor, R.; Csányi, G. Gaussian approximation potentials: The accuracy of quantum mechanics, without the electrons. *Phys. Rev. Lett.* **2010**, *104*, 136403.
(25) Rupp, M.; Tkatchenko, A.; Müller, K.-R.; von Lilienfeld, O. A. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys. Rev. Lett.* **2012**, *108*, No. 058301.
(26) Ramakrishnan, R.; Dral, P. O.; Rupp, M.; von Lilienfeld, O. A. Big data meets quantum chemistry approximations: The $\Delta$ -machine learning approach. *J. Chem. Theory Comput.* **2015**, *11*, 2087−2096.
(27) Chmiela, S.; Tkatchenko, A.; Sauceda, H. E.; Poltavsky, I.; Schütt, K. T.; Müller, K.-R. Machine learning of accurate energy-conserving molecular force fields. *Sci. Adv.* **2017**, *3*, No. e1603015.
(28) Schütt, K.; Kindermans, P.-J.; Felix, H. E. S.; Chmiela, S.; Tkatchenko, A.; Müller, K.-R. SchNet: A continuous-filter convolutional neural network for modeling quantum interactions. *Adv. Neural Inf. Process. Syst.* **2017**, 992−1002.
(29) Smith, J. S.; Isayev, O.; Roitberg, A. E. ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chem. Sci.* **2017**, *8*, 3192−3203.
(30) Han, J.; Zhang, L.; Car, R. E. W. Deep Potential: a general representation of a many-body potential energy surface. *Commun. Comput. Phys.* **2018**, *23*, 629−639.
(31) Zhang, L.; Han, J.; Wang, H.; Car, R.; E, W. Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics. *Phys. Rev. Lett.* **2018**, *120*, 143001.
(32) Zhang, L.; Han, J.; Wang, H.; Saidi, W.; Car, R.; E, W. End-to-end symmetry preserving inter-atomic potential energy model for finite and extended systems. *Adv. Neural Inf. Process. Syst.* **2018**, 4436−4446.
(33) Brockherde, F.; Vogt, L.; Li, L.; Tuckerman, M. E.; Burke, K.; Müller, K.-R. Bypassing the Kohn-Sham equations with machine learning. *Nat. Commun.* **2017**, *8*, 872.
(34) Grisafi, A.; Fabrizio, A.; Meyer, B.; Wilkins, D. M.; Corminboeuf, C.; Ceriotti, M. Transferable machine-learning model of the electron density. *ACS Cent. Sci.* **2018**, *5*, 57−64.
(35) Chandrasekaran, A.; Kamal, D.; Batra, R.; Kim, C.; Chen, L.; Ramprasad, R. Solving the electronic structure problem with machine learning. *npj Comput. Mater.* **2019**, *5*, 1−7.
(36) Zepeda-Núñez, L.; Chen, Y.; Zhang, J.; Jia, W.; Zhang, L.; Lin, L. Deep Density: circumventing the Kohn-Sham equations via symmetry preserving neural networks. 2019, arXiv:1912.00775. *arXiv.org*. https://arxiv.org/abs/1912.00775.
(37) Friedman, J. H. Greedy function approximation: a gradient boosting machine. *Ann. Math. Stat.* **2001**, 1189−1232.
(38) Sun, Q.; Berkelbach, T. C.; Blunt, N. S.; Booth, G. H.; Guo, S.; Li, Z.; Liu, J.; McClain, J. D.; Sayfutyarova, E. R.; Sharma, S.; et al. PySCF: the Python-based simulations of chemistry framework. *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **2018**, *8*, e1340.
(39) Sauceda, H. E.; Chmiela, S.; Poltavsky, I.; Müller, K.-R.; Tkatchenko, A. Molecular force fields with gradient-domain machine learning: Construction and application to dynamics of small molecules with coupled cluster forces. *J. Chem. Phys.* **2019**, *150*, 114102.
(40) Cheng, L.; Welborn, M.; Christensen, A. S.; Miller, T. F., III *Thermalized (350K) QM7b, GDB-13, water, and short alkane quantum chemistry dataset including MOB-ML features*. https://data.caltech.edu/records/1177 (accessed July 7, 2020)
(41) Cheng, L.; Kovachki, N. B.; Welborn, M.; Miller, T. F., III Regression Clustering for Improved Accuracy and Training Costs with


https://dx.doi.org/10.1021/acs.jctc.0c00872
*J. Chem. Theory Comput.* 2021, 17, 170−181

Molecular-Orbital-Based Machine Learning. *J. Chem. Theory Comput.* **2019**, *15*, 6668−6677.
(42) Christensen, A. S.; Bratholm, L. A.; Faber, F. A.; Anatole von Lilienfeld, O. FCHL revisited: Faster and more accurate quantum machine learning. *J. Chem. Phys.* **2020**, *152*, No. 044107.
(43) Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; Desmaison, A.; Kopf, A.; Yang, E.; DeVito, Z.; Raison, M.; Tejani, A.; Chilamkurthy, S.; Steiner, B.; Fang, L.; Bai, J.; Chintala, S. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *Adv. Neural Inf. Process. Syst.* **2019**, 8024−8035.
(44) Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. 2014, arXiv:1412.6980. *arXiv.org*. https://arxiv.org/abs/1412.6980.
(45) Stukowski, A. Visualization and analysis of atomistic simulation data with OVITO−the Open Visualization Tool. *Modell. Simul. Mater. Sci. Eng.* **2010**, *18*, No. 015012.
(46) Smith, J. S.; Nebgen, B.; Lubbers, N.; Isayev, O.; Roitberg, A. E. Less is more: Sampling chemical space with active learning. *J. Chem. Phys.* **2018**, *148*, 241733.
(47) Smith, J. S.; Nebgen, B. T.; Zubatyuk, R.; Lubbers, N.; Devereux, C.; Barros, K.; Tretiak, S.; Isayev, O.; Roitberg, A. E. Approaching coupled cluster accuracy with a general-purpose neural network potential through transfer learning. *Nat. Commun.* **2019**, *10*, 2903.
(48) Peverati, R.; Zhao, Y.; Truhlar, D. G. Generalized gradient approximation that recovers the second-order density-gradient expansion with optimized across-the-board performance. *J. Phys. Chem. Lett.* **2011**, *2*, 1991−1997.
(49) Luo, S.; Zhao, Y.; Truhlar, D. G. Validation of electronic structure methods for isomerization reactions of large organic molecules. *Phys. Chem. Chem. Phys.* **2011**, *13*, 13683−13689.
(50) Momma, K.; Izumi, F. *VESTA 3* for three-dimensional visualization of crystal, volumetric and morphology data. *J. Appl. Crystallogr.* **2011**, *44*, 1272−1276.


https://dx.doi.org/10.1021/acs.jctc.0c00872
*J. Chem. Theory Comput.* 2021, 17, 170−181