





RESEARCH ARTICLE | SEPTEMBER 06 2023

Construct exchange-correlation functional via machine learning

Jiang Wu; Sai-Mang Pun ; Xiao Zheng ; GuanHua Chen  



J. Chem. Phys. 159, 090901 (2023)

<https://doi.org/10.1063/5.0150587>



View
Online



Export
Citation

CrossMark

Construct exchange-correlation functional via machine learning

Cite as: J. Chem. Phys. 159, 090901 (2023); doi: 10.1063/5.0150587

Submitted: 15 March 2023 • Accepted: 27 July 2023 •

Published Online: 6 September 2023



View Online



Export Citation



CrossMark

Jiang Wu,^{1,2} Sai-Mang Pun,^{1,2}  Xiao Zheng,³  and GuanHua Chen^{1,2,a)} 

AFFILIATIONS

¹ Department of Chemistry, The University of Hong Kong, Hong Kong, China

² Hong Kong Quantum AI Lab, Hong Kong, China

³ Department of Chemistry, Fudan University, Shanghai 200433, China

^{a)} Author to whom correspondence should be addressed: gbc@everest.hku.hk

ABSTRACT

Density functional theory has been widely used in quantum mechanical simulations, but the search for a universal exchange-correlation (XC) functional has been elusive. Over the last two decades, machine-learning techniques have been introduced to approximate the XC functional or potential, and recent advances in deep learning have renewed interest in this approach. In this article, we review early efforts to use machine learning to approximate the XC functional, with a focus on the challenge of transferring knowledge from small molecules to larger systems. Recently, the transferability problem has been addressed through the use of quasi-local density-based descriptors, which are rooted in the holographic electron density theorem. We also discuss recent developments using deep-learning techniques that target high-level *ab initio* molecular energy and electron density for training. These efforts can be unified under a general framework, which will also be discussed from this perspective. Additionally, we explore the use of auxiliary machine-learning models for van der Waals interactions.

Published under an exclusive license by AIP Publishing. <https://doi.org/10.1063/5.0150587>

I. INTRODUCTION

In 1964, Hohenberg and Kohn proved the unique mapping between the ground state electron density and local potential, besides an overall constant.¹ This insight led to the Kohn–Sham formulation of density functional theory and the notion of exchange-correlation (XC) energy functional, introduced by Kohn and Sham in 1965.² The Kohn–Sham approach provides a way to transform the many-electron problem into an equivalent one-electron problem with an effective potential. The search for the universal XC functional has resulted in a variety of approximate XC functionals since then. However, the universal XC energy functional has remained elusive. Although a universal analytical form for the XC functional is believed to be impractical, the search for a universal XC functional remains an active area of research. For the state-of-the-art of DFT, we refer to the reader to Ref. 3.

Machine learning (ML) has been applied to construct the XC functional in DFT since 1996 when Handy *et al.* proposed a machine learning approach to map the local electron density to the local XC potential.⁴ In 2004, Zheng *et al.* independently used a neural

network to construct an improved XC energy functional based on the functional form of B3LYP.⁵ With the success of deep learning in computer vision,⁶ natural language processing,⁷ and other fields,^{8,9} there is growing interest in using deep learning algorithms, such as convolution neural networks (CNNs),¹⁰ graph neural networks (GNNs),¹¹ and transformers,¹² to approximate the universal XC functional.

Specifically, efforts to develop machine-learning based (MLB) XC functional or potential can be categorized into several types: (i) MLB XC potential,^{13,14} (ii) MLB XC energy functional,^{5,15–27} (iii) MLB XC energy density,¹⁶ and (iv) MLB XC energies of fragments.^{27–39} In addition to the XC functional or potential, other aspects of the DFT framework can also benefit from ML techniques. For instance, the kinetic energy functional has been proposed.^{40–48} Furthermore, ML has been extensively used to fit or construct potential energy surfaces,⁴⁹ where DFT is frequently used as a training target or benchmark for the ML algorithms. Besides the above-mentioned works on MLB XC functionals, researchers have employed other data-driven techniques rather than deep learning (such as genetic algorithms) to seek accurate forms of XC

functionals.^{50,51} For example, in Ref. 50, the authors have proposed a Symbolic Functional Evolutionary search to construct accurate XC functionals in the symbolic form.

In this perspective, we focus on the construction of MLB XC functional or potential and review various methodologies, from the early approaches to the latest developments.⁵² It is important to note that our goal in this article is not to be exhaustive. Rather, we aim to explore specifically how machine learning techniques can be used to construct XC functionals or potentials and answer the question of whether the universal functional can be accurately obtained via deep learning.

II. EXCHANGE-CORRELATION FUNCTIONAL AND POTENTIAL

Our discussion begins with an introduction to the fundamental concepts of the DFT framework that will be referenced throughout the subsequent sections. The Hohenberg–Kohn theorem¹ forms the basis for predicting the quantum mechanical properties of a many-electron system from its electron density, implying that the ground state energy is a unique functional of the electron density (denoted as $\rho(\mathbf{r})$). By introducing a non-interacting reference system, Kohn and Sham² expressed the ground state energy functional $E[\rho(\mathbf{r})]$ as follows:

$$\begin{aligned} E[\rho(\mathbf{r})] &= T_s[\rho(\mathbf{r})] + E_{\text{ext}}[\rho(\mathbf{r})] + E_H[\rho(\mathbf{r})] + E_{\text{XC}}[\rho(\mathbf{r})], \\ &= -\frac{1}{2} \sum_{i=1}^{\text{occ}} \langle \phi_i | \nabla^2 | \phi_i \rangle + \int d\mathbf{r} \rho(\mathbf{r}) v_{\text{ext}}(\mathbf{r}) \\ &\quad + \frac{1}{2} \int d\mathbf{r} d\mathbf{r}' \frac{\rho(\mathbf{r})\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + E_{\text{xc}}[\rho(\mathbf{r})], \end{aligned} \quad (1)$$

and

$$E_{\text{xc}}[\rho(\mathbf{r})] = T[\rho(\mathbf{r})] - T_s[\rho(\mathbf{r})] + E_{\text{ee}}[\rho(\mathbf{r})] - E_H[n]. \quad (2)$$

Here, T_s , E_{ext} , E_H , and E_{xc} stand for the Single-Slater kinetic energy with a set of orbitals $\{\phi_i\}$, the external energy, the Hartree energy, and the exchange correlation energy, respectively. The terms T and E_{ee} are the exact kinetic energy and the Coulomb energy for the many-electron interacting system, respectively. Minimizing the total energy constrained by normalized orbitals leads to the following Kohn–Sham (KS) equations:

$$\left(-\frac{1}{2} \nabla^2 + v_{\text{ext}}(\mathbf{r}) + \int d\mathbf{r}' \frac{\rho(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} + v_{\text{xc}}(\mathbf{r}) \right) \phi_i(\mathbf{r}) = \varepsilon_i \phi_i(\mathbf{r}), \quad (3)$$

where v_{xc} is called the exchange-correlation potential, which is the functional derivative of the exchange-correlation energy with respect to the electron density,

$$v_{\text{xc}}(\mathbf{r}) = \frac{\delta E_{\text{xc}}[\rho(\mathbf{r})]}{\delta \rho(\mathbf{r})}.$$

The left-hand side of the Kohn–Sham Eq. (3) includes the electron density ρ (depending on the orbitals ϕ_i 's), and thus, it is a nonlinear eigenvalue problem. To solve this problem, an initial density ρ_0 must be provided, and the solution must be updated until convergence is reached, a process known as self-consistent field (SCF) calculation.⁵³

A key starting point for using ML techniques within DFT is to parameterize the XC energy functional (or potential, or even the corresponding energy density) using various ML architectures, such as neural networks,⁵⁴ and train the model with carefully designed descriptors for input and training data. This is referred to as the ML-DFT method, and the ML architecture is termed the ML-DFT model in this perspective. Descriptors should be the functions or functionals of electron density. Below, we review the existing ML-DFT methodologies based on different types of descriptors used in modeling.

III. EARLY WORKS ON XC MODELS

Prior to the recent research to construct XC functional or potential via ML, two research groups employed neural networks to search for the XC functional and potential, and the two pioneering publications^{4,5} were published in 1996 and 2004, respectively. The electron density was used for the descriptors, and the output was the XC potential or functional.

A. Neural network-based B3LYP functional

As one of the most popular hybrid functionals, the B3LYP functional⁵⁵ includes five pure functional terms: (i) the Slater exchange functional $E_X^{\text{Slater}}[\rho]$;⁵⁶ (ii) the Hartree–Fock exchange functional $E_X^{\text{HF}}[\rho]$;⁵⁷ (iii) the difference between the Becke88 exchange⁵⁸ and the Slater functionals, denoted as $\Delta E_X^{\text{Becker}}[\rho] = E_X^{\text{B88}}[\rho] - E_X^{\text{Slater}}[\rho]$; (iv) the Lee–Yang–Parr correlation functional E_C^{LYP} ;⁵⁵ and (v) the Vosko–Wilk–Nusair correlation functional E_C^{VWN} .⁵⁹ The B3LYP functional is tuned by three coefficients: a_0 , a_X , and a_C ; it reads as follows:

$$\begin{aligned} E_{\text{xc}}^{\text{B3LYP}}[\rho] &= a_0 E_X^{\text{Slater}}[\rho] + (1 - a_0) E_X^{\text{HF}}[\rho] + a_X \Delta E_X^{\text{Becker}}[\rho] \\ &\quad + a_C E_C^{\text{LYP}}[\rho] + (1 - a_C) E_C^{\text{VWN}}[\rho]. \end{aligned} \quad (4)$$

In hybrid functionals like B3LYP, the coefficients are typically determined by fitting to experimental data or accurate calculations, and once obtained, they are treated as constants. In B3LYP, the values are $a_0 = 0.8$, $a_X = 0.72$, and $a_C = 0.81$, based on fitting a set of atomization energies and ionization potential.⁵⁸ See also Ref. 60 for calibration and selection of hybrid density functionals using Bayesian optimization techniques.

In 2004, Zheng *et al.*⁵ proposed to project the exact XC functional onto the B3LYP functional and pointed out that a_0 , a_X , and a_C should, in theory, be system-dependent or functional of electron density. By making these coefficients as functionals of density, the exact XC functional can be expressed as

$$\begin{aligned} E_{\text{xc}}^{\text{Exact}}[\rho] &= a_0[\rho] E_X^{\text{Slater}}[\rho] + (1 - a_0[\rho]) E_X^{\text{HF}}[\rho] \\ &\quad + a_X[\rho] \Delta E_X^{\text{Becker}}[\rho] + a_C[\rho] E_C^{\text{LYP}}[\rho] \\ &\quad + (1 - a_C[\rho]) E_C^{\text{VWN}}[\rho], \end{aligned} \quad (5)$$

and the resulting coefficients become clearly system-dependent, with different values for different density inputs. Thus, learning the density functional coefficients is essential for determining the exact density functional. In an effort to learn such XC functional, Zheng *et al.*⁵ proposed a neural network with five descriptors as inputs a single hidden layer. The outputs of the ML model are the coefficients

$a_0[\rho]$, $a_X[\rho]$, and $a_C[\rho]$. The resulting XC functional is used in the KS-SCF calculations. Next, we briefly discuss the computation of the

XC potential used in the SCF calculation. The exact XC potential reads as follows:

$$v_{xc}^{\text{Exact}} = \frac{\delta E_{xc}^{\text{B3LYP}}}{\delta \rho(\mathbf{r})} = a_0[\rho] \underbrace{\frac{\delta E_X^{\text{Slater}}}{\delta \rho(\mathbf{r})} + (1 - a_0[\rho]) \frac{\delta E_X^{\text{HF}}}{\delta \rho(\mathbf{r})} + a_X[\rho] \frac{\delta \Delta E_X^{\text{Becker}}}{\delta \rho(\mathbf{r})} + a_C[\rho] \frac{\delta E_C^{\text{LYP}}}{\delta \rho(\mathbf{r})} + (1 - a_C[\rho]) \frac{\delta E_C^{\text{VWN}}}{\delta \rho(\mathbf{r})}}_{\text{terms containing derivatives w.r.t. energy functionals}} + \underbrace{\frac{\delta a_0[\rho]}{\delta \rho(\mathbf{r})} E_X^{\text{Slater}} - \frac{\delta a_0[\rho]}{\delta \rho(\mathbf{r})} E_X^{\text{HF}} + \frac{\delta a_X[\rho]}{\delta \rho(\mathbf{r})} \Delta E_X^{\text{Becker}} + \frac{\delta a_C[\rho]}{\delta \rho(\mathbf{r})} E_C^{\text{LYP}} - \frac{\delta a_C[\rho]}{\delta \rho(\mathbf{r})} E_C^{\text{VWN}}}_{\text{terms containing derivatives w.r.t. coefficients}}. \quad (6)$$

When one assumes that those coefficients do not depend on ρ too much, that is,

$$\frac{\delta a_0[\rho]}{\delta \rho(\mathbf{r})} \approx \frac{\delta a_X[\rho]}{\delta \rho(\mathbf{r})} \approx \frac{\delta a_C[\rho]}{\delta \rho(\mathbf{r})} \approx 0,$$

the potential can be (approximately) written as follows:

$$v_{xc} \approx a_0[\rho] \frac{\delta E_X^{\text{Slater}}}{\delta \rho(\mathbf{r})} + (1 - a_0[\rho]) \frac{\delta E_X^{\text{HF}}}{\delta \rho(\mathbf{r})} + a_X[\rho] \frac{\delta \Delta E_X^{\text{Becker}}}{\delta \rho(\mathbf{r})} + a_C[\rho] \frac{\delta E_C^{\text{LYP}}}{\delta \rho(\mathbf{r})} + (1 - a_C[\rho]) \frac{\delta E_C^{\text{VWN}}}{\delta \rho(\mathbf{r})}. \quad (7)$$

With the above approximation in Formula (7), the machine-learned XC potential was trained and tested in SCF calculations for 116 small molecules, yielding improvements over the original B3LYP functional. Using the basis set of 6-311+G(3df,2p), the RMS errors in overall energies using the conventional B3LYP is 4.7 kcal mol⁻¹, while the NN-based functional gives 2.9 kcal mol⁻¹ (see Table I below). However, the resulting MLB XC functional is not as accurate due to the above approximation that the functional derivatives of a_0 , a_X , and a_C are zero.

TABLE I. Performance of the ML functional. Reprinted with permission from Zheng *et al.*, Chem. Phys. Lett. **390**(1–3), 186–192 (2004). Copyright (2004) Elsevier. AE: Atomization Energy; IP: Ionization Potential; PA: Proton Affinity; and TAE: Total Atomic Energy.

RMS errors (all data are in the units of kcal mol ⁻¹)					
Properties	AE	IP	PA	TAE	Overall
Number of samples	56	42	8	10	116
A ^a	2.9	3.9	1.9	4.1	3.4
DFT-1 ^b	3.0	4.9	1.6	10.3	4.7
DFT-NN ^c	2.4	3.7	1.6	2.7	2.9

^aBecke's work.

^bConventional B3LYP/6-311+G(3df, 2p).

^cNeural-Networks-based B3LYP/6-311+G(3df, 2p).

B. Neural network-based XC potential model

In 1996, Tozer *et al.*⁴ proposed a neural network architecture that mapped local electron density to the corresponding local XC potential. The method is classified as *local descriptor-based* due to its single density input. It is expected to achieve improvements if the information from higher order derivatives of densities is employed. In Ref. 4, the input densities were calculated at the CCSD level (with Brueckner coupled cluster method⁶¹) and the model consisted of one fully connected layer with eight hidden neurons; while the target XC potentials were computed by the Zhao–Morrison–Parr (ZMP) method.⁶² Trainings were performed on (ρ, v_{xc}) pairs from either one molecule or multiple atoms/molecules. The ML-DFT model was used to perform KS-SCF calculations, resulting in significant improvements over LDA (see the column of CNN in Table II for the numerical performance of the method). These improvements can be enhanced further by including more information from the neighboring area of the local point, for instance, by adding first and

TABLE II. The calculated ionization potentials are compared with that of LDA as well as experimental values. Remark: CNN stands for *Computational Neural Network* instead of the *Convolution Neural Network*. Reproduced with permission from Tozer *et al.*, J. Chem. Phys. **105**, 9200–9213 (1996). Copyright 1996 AIP Publishing LLC.

	LDA	CNN	$-I$
Ne ^a	-0.492	-0.660	-0.792
HF ^a	-0.350	-0.525	-0.590
N ₂ ^a	-0.380	-0.560	-0.573
H ₂ O ^a	-0.261	-0.441	-0.463
H ₂ ^a	-0.369	-0.550	-0.567
CO	-0.333	-0.519	-0.515
F ₂	-0.347	-0.516	-0.577
CH ₄	-0.346	-0.535	-0.460
NH ₃	-0.222	-0.404	-0.373
C ₂ H ₂	-0.270	-0.461	-0.419
O ₃	-0.293	-0.468	-0.457
LiH	-0.159	-0.422	-0.283
Li ₂	-0.120	-0.394	-0.188

^aIn the training set.

higher-order derivatives of electron density in the descriptors, as pointed out by Tozer *et al.* in Ref. 4.

To summarize, the approach developed in Ref. 5 pioneers the research direction on constructing the XC functional using machine learning while the approach developed in Ref. 4 is the first work targeting directly the XC potential. The method in Ref. 4 uses information from local electron densities as the descriptor and we term it as the local descriptor-based method,⁶³ and obviously, it is only an approximation. The numerical scheme constructed in Ref. 5 intends to use the entire electron density of a molecule as the descriptors, and we term it as the global descriptor-based method. The global descriptor-based method can be, in principle, exact. In Sec. IV, we review, first, the recent studies on global descriptor-based methods utilizing more advanced machine-learning architectures.

IV. RECENT WORKS ON GLOBAL DESCRIPTOR-BASED XC MODELS

A. Deep neural networks for XC potential

The work by Nagai *et al.*¹³ investigated the idea of incorporating a neural-network trained XC potential model in the KS-SCF calculation. Specifically, this approach makes use of a fixed grid with 100 consecutive and equally spaced points to feed the entire density as a vector to a fully connected neural network with two 300-neuron hidden layers, mapping the entire electron density to the

target XC potential [see Fig. 1 (left) for the algorithmic procedure of the numerical scheme]. Once the XC potential model is trained and established, one can solve the Kohn–Sham equation, and the initial XC potential is produced via the neural-network trained XC potential model with the initial electron density as the descriptor. The total energy can also be evaluated.

The proposed method was tested in a 1-D model system consisting of two interacting spinless fermions with various random Gaussian external potentials. The target potential was set to be the total Coulomb potential $v_{Hxc} = v_H + v_{xc}$ with $v_H = -A \exp(-x^2/B^2)$ being the Hartree potential with two parameters A and B ; the corresponding density was calculated using exact diagonalization.

In Fig. 1 (right), the two columns show (as color maps) the out-of-training error in density and total energy derived from the KS scheme with the trained potentials. The horizontal and vertical axes represent the ranges of the parameters A and B , respectively. Overall, the trained neural network model demonstrated good generalizability in out-of-sample tests with unseen external potentials within the simple setup.

B. Projection-based XC potential and energy model

To simplify and standardize the density descriptor in realistic systems, a projection method may be chosen. In a recent

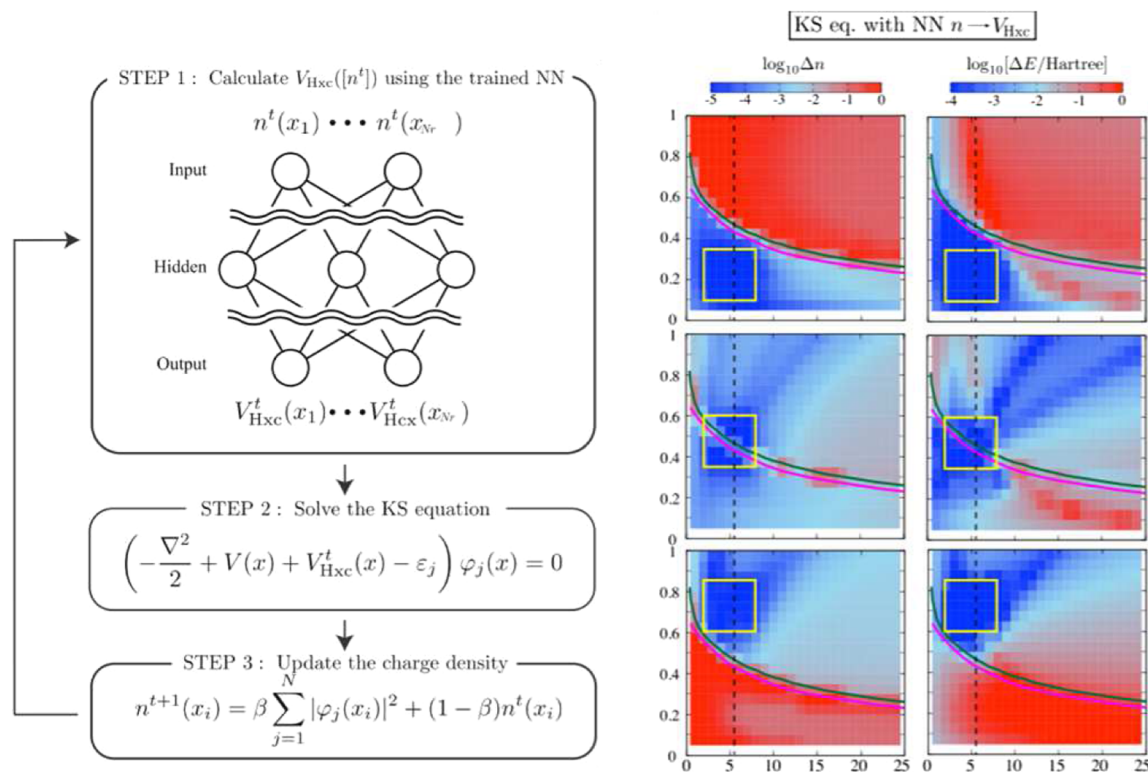


FIG. 1. Left: structure of the ML-DFT model developed in Ref. 13. Right: prediction error. Δn and ΔE represent the errors of SCF density and total energy with respect to the exact reference, respectively. Reproduced with permission from Nagai *et al.*, J. Chem. Phys. **148**, 241737 (2018). Copyright 2018 AIP Publishing LLC.

study, Bogojeski *et al.*²⁸ employed a machine learning method to predict the DFT or CCSD energies (or the correction to a standard DFT calculation) from DFT densities. In principle, they utilized a periodic Fourier basis set comprising 12 500 functions to perform a projection and represent each molecular density as follows:

$$\rho[v](\mathbf{r}) = \sum_{\ell=1}^L u^{(\ell)}[v]\phi_{\ell}(\mathbf{r}).$$

Here, $u^{(\ell)}$ and ϕ_{ℓ} are the ℓ th projection coefficient and ℓ th basis function, respectively. The term v denotes the external nuclear potential, which was approximated by a sum of Gaussians as in Ref. 28. The projection coefficient vector $\mathbf{u} = (u^{(\ell)})_{\ell=1}^L$ is then mapped to the target energy through the kernel ridge regression (KRR) model.⁶⁴ See Fig. 2(a) for its algorithmic procedure. The target energy was selected to be either the DFT energy obtained using the PBE functional, the CCSD(T) energy, or the difference between the two, which captures the exchange-correlation contribution at varying levels of accuracy.

The basic idea behind a KRR model is that when presented with a new density, the algorithm performs an interpolation based on known target energies from densities in the training set. The output of the model is written as

$$E_{\text{ML}}[\rho] = \sum_{i=1}^N \alpha_i k(\mathbf{u}[v], \mathbf{u}[v_i]), \quad (8)$$

where $k(\cdot, \cdot)$ is a Gaussian kernel measuring the similarity between any two projected density descriptor vectors,

$$k(\mathbf{u}, \mathbf{u}') = \exp\left(-\frac{|\mathbf{u} - \mathbf{u}'|^2}{2\sigma^2}\right),$$

with σ being a hyper-parameter determined by cross-validation.⁶⁵ In Eq. (8), the model E_{ML} stands for the fitted energy; $\mathbf{u}[v]$ denotes the projection coefficient vector for the external potential v ; and v_i is the i th external potential in the training set. Predictions for new densities are generated by a summation of parameters weighted by the

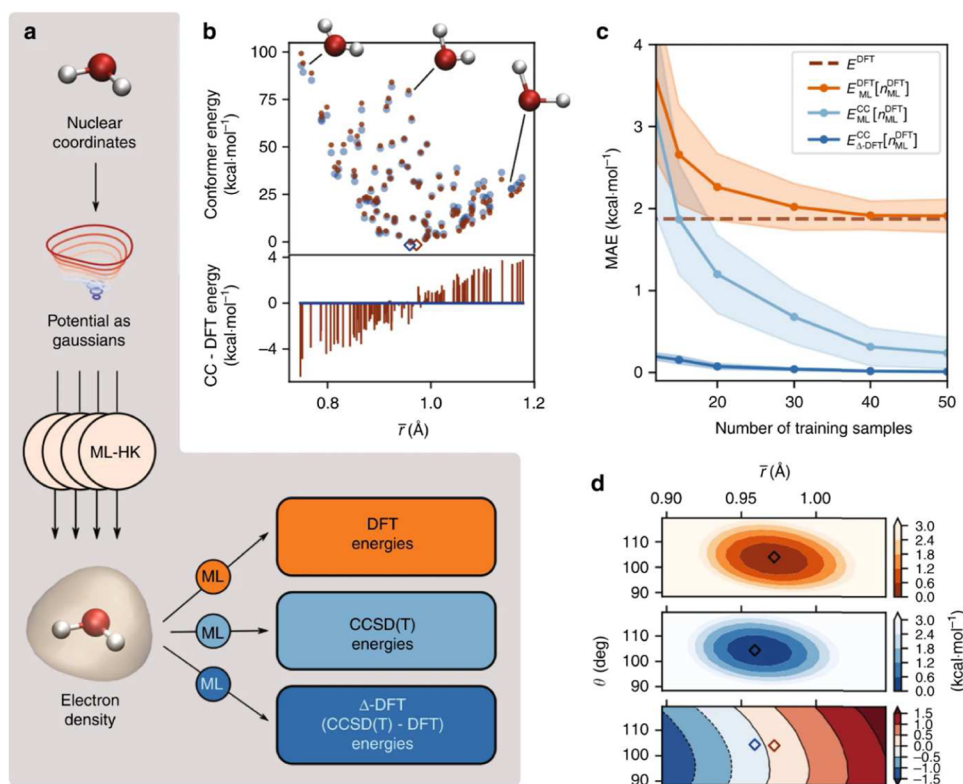


FIG. 2. (a) The KRR model constructed to represent the density functional, mapping the electron to either the DFT/CCSD(T) energy or their energy difference; another KRR ML model (ML-HK) was used to map the external potential to the density. (b) Energies (dark blue for CCSD(T) and dark orange for DFT (PBE)) of different water geometries in the training set. (c) Test set (other water geometries than the training set) MAE improves when the training set size increases. (d) The learned DFT (top), CCSD(T) (middle), and energy difference (bottom) surfaces, respectively. In (b) and (d), diamond scatter represents minimum energy geometries. Reproduced with permission from Bogojeski *et al.*, Nat. Commun. 11, 5223 (2020). Copyright 2020 Author(s), licensed under a Creative Commons Attribution 4.0 International License: <http://creativecommons.org/licenses/by/4.0/>.²⁸

kernel function, effectively representing unknown densities by interpolating known densities, with the interpolation parameters learned by the model.

To broaden the scope of their approach, the authors also built a separate KRR model mapping external potential to density. By combining this with the model that maps density to energy, all the density functionals can be expressed as functionals of external potential. This effectively blurs the line between machine learning methods based on density functional theory and those that directly learn from molecular geometry.

C. Kohn–Sham regularizer

Previous efforts have been made to construct XC potential models for SCF calculations. However, in those efforts, the training procedure and the SCF calculations were independent of each other. In contrast, in the work by Li *et al.*,²¹ the ML model was programmed in a fully differentiable way with the aid of automatic differentiation,^{66–70} allowing error to backpropagate through multiple iterations of the SCF calculation. In general, automatic differentiation allows efficiently computing the derivatives of any functions in the computer program, and this technique can be used, for instance, to minimize the Hartree–Fock energy (or any other objective functionals) to avoid eigenvalue calculation in an orbital-free setting.⁶⁸

The scheme developed by Li *et al.*²¹ effectively included more information about the functional mapping from the density to the XC energy, and the scheme was named the *Kohn–Sham regularizer*

(KSR) due to its generalization (preventing overfitting) capability (see also Ref. 71 for a spin-adapted version of KSR model). Figure 3(a) depicts the computational procedure of the KSR model, which uses the electron density of the molecule as the model input. Then, the model consists of a fixed number of times (denoted K as the total number of iterations) of SCF iterations [see Fig. 3(b) for the internal process of the SCF iteration], where each SCF iteration is parameterized by a neural network model, whose architecture is sketched in Fig. 3(c), outputting a series of energies $\{E_k\}_{k=1}^K$ and $\{\rho_k\}_{k=1}^K$. The loss function includes both energy and density loss terms. All the terms E_k 's are used to form the energy loss function while only the last term ρ_K in the charge density sequence will be used to form the density loss function. In other words, the former loss had contributions from multiple iterations, with decay weights for earlier iterations, while the latter only contains the root mean squared error between the last iteration's output and the target.

The KSR model has shown the generalizability for a simple one-dimensional H_2 model system, with only two training examples needed to determine the whole dissociation curve reasonably well. However, as the work was developed for 1D model systems, it still falls under the category of *proof-of-concept*. Moreover, the energy loss term contains the contributions of all produced energies E_k 's from the previous SCF iterations, and this training mechanism enforces that the output energy of the model should converge more or less exactly in the way the training labels did, which is generally not practical in the conventional SCF calculations. Extending it to realistic 3D systems will require extra effort due to the computational complexity.

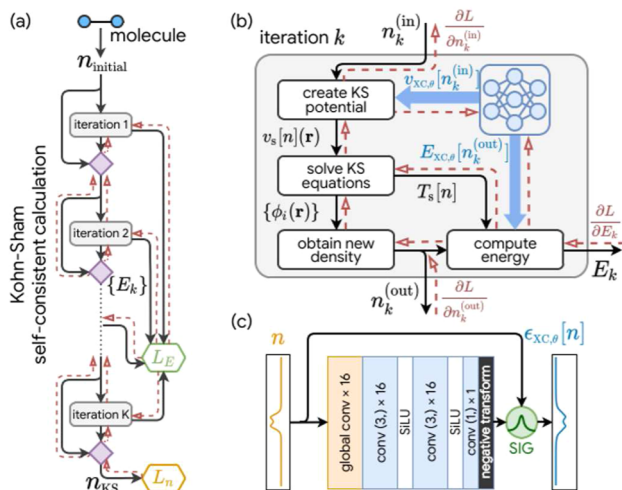


FIG. 3. The structure of ML XC energy model by Li *et al.*²¹ that includes KS SCF in the training. The forward and backward propagations pass in SCF during training are depicted in (a) as black solid and red dashed lines, respectively. (b) The details of one iteration of SCF with the parameterized neural network. The structure that utilizes the quasi-local information of the density to produce the XC energy density is depicted in (c). Instead of ρ , the symbol n is used for density, which is consistent with the symbol used in the original work. Reproduced with permission from Li *et al.*, Phys. Rev. Lett. **126**, 036401 (2021). Copyright 2021 Author(s), licensed under a Creative Commons Attribution 4.0 International License: <http://creativecommons.org/licenses/by/4.0/>.²¹

V. MODEL TRANSFERABILITY AND HOLOGRAPHIC ELECTRON DENSITY THEOREM

Although the application of high precision quantum chemistry methods, such as CCSD⁷² and quantum Monte Carlo,^{73,74} facilitates the acquisition of large amounts of data on small molecules, obtaining such an accurate dataset for large molecules from *ab initio* methods is not practical. The lack of such data for larger molecules poses a key problem to the transferability of machine-learning-based XC functionals of complex molecules. Since most of the existing ML-DFT models are trained only with the datasets of small molecules, the model's transferability, from simple and small molecules to complicated and large ones, may pose a challenge in constructing a universal XC functional. To address this issue, the density descriptors must be carefully designed to ensure the transferability of the ML-DFT model from small molecules to large ones.

Riess and Münch⁷⁵ posited in 1981 that the electron density distribution of a molecular system is determined by an arbitrary finite volume of the ground state electron density, based on the hypothesis that electron density functions of atomic and molecular species are real analytic in real space excluding nuclei. The validity of this hypothesis, however, was not rigorously proven until Fournais *et al.* demonstrated the real analyticity of electron density of arbitrary atomic and molecular eigenstate of the Schrödinger Equation.^{76,77} Another proof of real analyticity of electron density has been given by Jecko.⁷⁸ The ground state holographic electron

density theorem (GS-HEDT) named by Mezey⁷⁹ is thought to be linked to the concept of quantum similarity measures in DFT.^{80,81}

In the case of an atomic and molecular system, the external potential $v(\mathbf{r})$ acting on each electron is real analytic (mathematically defined) except at the nuclei. The electron density is real analytic everywhere except for isolated points where the nuclei's point charges cause non-analyticities. Analytic functions, such as Gaussian orbitals and plane waves, are often used as basis sets for quantum mechanical calculations, resulting in real analytic electron densities. The values within a subregion is sufficient to determine values everywhere in the physical space, and this can be shown by the analytic continuation of real analytic functions, as demonstrated in Ref. 82. Zheng *et al.* have provided a simple proof for the holographic property of real analytic density in three-dimensional physical space and proposed the time-dependent holographic electron density theorem for open electronic systems, which has been applied to the study of time-dependent quantum transport problems.^{83–86} Moreover, the *nearsightedness principle* proposed by Kohn⁸⁷ (see also Ref. 88) suggests that local electronic properties, such as the electron density, depend mostly on the external potential in the nearby regions. This principle shares the same foundation with the GS-HEDT, which also highlights the local nature of ground state electrons.

Based on the GS-HEDT, the electron density within a finite volume is sufficient to determine the global density distribution of a real atomic or molecular system. While many modern density functional approximations utilize nonlocal information for improved accuracy,⁸⁹ it may be possible to achieve an accurate quasi-local KS mapping through the use of advanced machine learning techniques,

$$v_{\text{xc}}[\rho](\mathbf{r}) = v_{\text{xc}}[\rho(\mathbf{r}') : |\mathbf{r} - \mathbf{r}'| < \delta, \forall \delta)0](\mathbf{r}). \quad (9)$$

To create an ML-DFT model for quasi-local electron density, a direct mapping of the electron density to the XC potential for use in the SCF calculations can be used as a starting point. We may write

$$v_{\text{ML-XC}}(\mathbf{r}) = M_{\theta}[\rho(\mathbf{r}')|_{r' \in B(\mathbf{r})}], \quad (10)$$

where M_{θ} denotes the ML-DFT model with its optimized parameters denoted as θ , and $B(\mathbf{r})$ denotes a neighborhood of \mathbf{r} . The ML XC potential $v_{\text{ML-XC}}(\mathbf{r})$ is dependent on the electron density ρ at \mathbf{r} and its neighborhood. After training, the resulting ML-DFT model for $v_{\text{ML-XC}}(\mathbf{r})$ can be used in SCF calculations. As dictated by the GS-HEDT, the neighborhood could be arbitrarily small in principle. However, in practice, the quasi-local region surrounding the spatial point should be of a certain finite size to ensure the numerically feasible KS mapping. In Ref. 14, a cube centered at each position with sampling points arranged along their spatial directions is a viable neighborhood choice. For instance, for a given window half-length $h > 0$, the sampling points range from the cube,

$$B(\mathbf{r}) = [r_x - h, r_x + h] \times [r_y - h, r_y + h] \times [r_z - h, r_z + h],$$

where $\mathbf{r} = (r_x, r_y, r_z)$ with a certain step length (the smaller the step, the more points are sampled given a fixed h). The output of the ML-DFT model is the value of XC potential at \mathbf{r} , and, therefore, once trained, the model predicts the XC potential at position \mathbf{r} of the center of the sampling neighborhood. The entire XC potential is obtained by sweeping the model across the grid, and the output is

used in the KS equation within the SCF procedure to calculate a new density.

The above ML-DFT model that uses the quasi-local electron density as the descriptors is termed the quasi-local descriptor-based XC model. In the next session, we review three different types of quasi-local descriptor-based ML-DFT models.

VI. QUASI-LOCAL DESCRIPTOR-BASED XC MODELS

Compared to the local descriptor-based ML-DFT model such as the one proposed in Ref. 5, the quasi-local descriptor-based model can, in principle, be exact and, in practice, is certainly more accurate. This is justified by the HEDT, which states that the ground state electron density uniquely determines the ground state properties of any subdomain and of the total domain of the system. The quasi-local descriptor-based ML-DFT methods are promising.

A. Quasi-local XC potential model

In Ref. 14, Zhou *et al.* proved the rigorous foundation of the quasi-local descriptor-based ML-DFT method and, in addition, developed and implemented its ML-DFT and subsequent KS-SCF algorithm. Quasi-local densities (input or descriptors) and XC potentials (labeled data) were discretized on a grid whose points coincide with the set of quadrature points for potential integration. A convolution neural network (CNN)⁹⁰ architecture was employed with the input being a cube of sampled density, and the final output of the model is a scalar value of the XC potential at the respective quadrature point. The resultant ML XC potential is integrated and used for SCF calculations later on.

The ML-DFT model is a 3D CNN neural network, as depicted in Fig. 4. It was tested on H_2 and HeH^+ and trained on a dataset of 50 H_2 molecules and 50 HeH^+ ions (with bond lengths ranging from 0.504 to 0.896 Å). The ground state electron density is used as the input or descriptor and was calculated by employing CCSD(T). The target or output is the XC potential, which was calculated using Wu-Yang method^{91,92} (see Appendix B for a brief introduction).

This ML-DFT model outperforms traditional DFT using B3LYP in terms of electron density accuracy by at least one order of magnitude, as demonstrated by benchmarking with the reference CCSD electron density. When integrated into the SCF procedure, the ML XC potential achieves impressive performance on the elec-

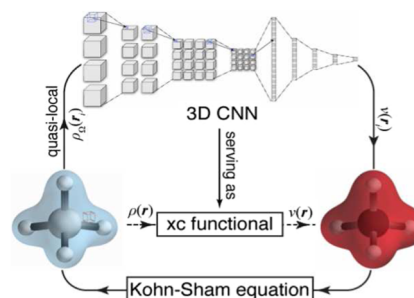


FIG. 4. Structure of the 3D CNN model for molecules discretized on grid points. Adapted with permission from Zhou *et al.*, *J. Phys. Chem. Lett.* **10**(22), 7264–7269 (2019). Copyright (2019) American Chemical Society.

tron density, surpassing B3LYP by up to two orders of magnitude. In Fig. 5(a), HeH⁺ electron density calculated with the ML-DFT method is compared with B3LYP, and the reference data are the CCSD(T) electron density. With the predicted electron density, an atomic force can be calculated using Hellman–Feynman theorem⁹³ and basis set correction.⁹⁴ The accuracy is significantly better than that of B3LYP.

Figure 5(b) shows that the same model was tested on HeH⁺ ions with He–H distances up to values much larger than those in the training set. The model's out-of-sample performance, as measured by the density difference to CCSD, remained much smaller than that of B3LYP even at bond distances around 3 Å for HeH⁺. Furthermore, the density performance of the ML-DFT model outperformed that of B3LYP even in more complex systems (such as He–H–H–He²⁺) with different numbers of electrons and nuclei than molecules in the training set, and Figs. 5(c) and 5(d) show the comparison of two different structures, respectively. The use of quasi-local electron density as input has yielded exceptional transferability of the ML-DFT model.

B. Quasi-local XC energy density model

An alternative approach is to build an ML-DFT model that directly targets the XC energy density ε_{xc} defined as follows:

$$E_{xc}[\rho] = \int d\mathbf{r} \varepsilon_{xc}(\mathbf{r})\rho(\mathbf{r}) = \int d\mathbf{r} \varepsilon_{xc}[\mathbf{r}, \rho(\mathbf{r}), B(\mathbf{r})]\rho(\mathbf{r}),$$

$$v_{xc}(\mathbf{r}, [\rho]) = \frac{\delta E_{xc}}{\delta \rho(\mathbf{r})} = \varepsilon_{xc}(\mathbf{r}) + \int d\mathbf{r}' \frac{\delta \varepsilon_{xc}(\mathbf{r}', \rho(\mathbf{r}'), B(\mathbf{r}))}{\delta \rho(\mathbf{r}')} \rho(\mathbf{r}'). \quad (11)$$

Although targeting the energy densities shares similarities with the previous ML-DFT models for the XC potential, the model

output is different, and careful consideration should be taken. Similar to the previous ML-DFT models, this model requires data in the form of XC potential or electron density at each grid point, and a sensible strategy is training targets for the entire grid. Unfortunately, unlike the XC potential, there is no procedure like the WY method^{91,92} to produce the energy density. Furthermore, the calculation of parameters requires second order derivatives, which can be computationally intensive. Nevertheless, automatic differentiation techniques and packages are now available to handle such calculations. Implementing the model involves saving the first derivative graph and including other numerical burdens in the backpropagation process to calculate second order derivatives. The XC energy and potential can be obtained via numerical manipulation from the XC energy density. To generate the total XC energy, the XC energy density can be integrated weighted by electron density.

Below, we present the ML-DFT model for the XC energy density developed by Nagai *et al.*,²⁴ which employs a fully-connected neural network model trained with different electron density descriptors as inputs and the XC energy density as output. However, the XC energy density is not directly used for training loss function (only losses in total energy and electron density are employed). The electron density descriptors used in the model include various combinations of the following density-related quantities:

$$\rho(\mathbf{r}), \quad \zeta(\mathbf{r}) = \frac{\rho_i(\mathbf{r}) - \rho_t(\mathbf{r})}{\rho(\mathbf{r})}, \quad s(\mathbf{r}) = \frac{|\rho(\mathbf{r})|}{2(3\pi^2)^{1/3} \rho^{4/3}(\mathbf{r})},$$

$$\tau(\mathbf{r}) = \frac{1}{2} \sum_{i=1}^{\text{occ}} |\nabla \varphi_i(\mathbf{r})|^2, \quad (12)$$

$$R(\mathbf{r}) = \int d\mathbf{r}' \rho(\mathbf{r}') \exp\left(-\frac{|\mathbf{r} - \mathbf{r}'|}{\sigma}\right).$$

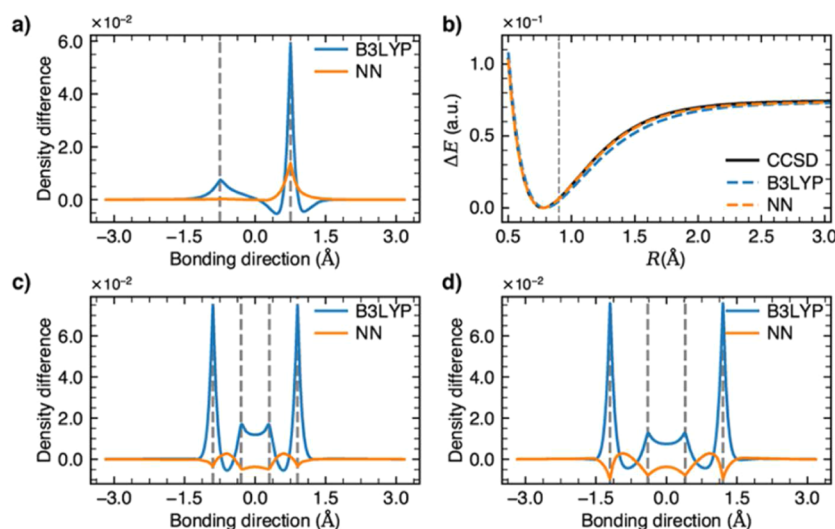


FIG. 5. Performance of the ML XC Potential model (the KS-DFT/NN model in Ref. 14) on HeH⁺ and He–H–H–He²⁺. (a) Density difference of HeH⁺; (b) relative energy of HeH⁺. The lowest energy is shifted to zero for comparison; (c) density difference of linear H₃⁺; and (d) density difference of He–H–H–He²⁺. Adapted with permission from Zhou *et al.*, J. Phys. Chem. Lett. **10**(22), 7264–7269 (2019). Copyright (2019) American Chemical Society.

Denote \mathbf{g} the overall input vector concatenating all necessary input descriptors. Then, the XC energy density is parameterized as

$$\varepsilon_{\text{xc}}[\rho](\mathbf{g}) = -\rho^{1/3} \left(\frac{(1+\zeta)^{4/3} + (1-\zeta)^{4/3}}{2} \right) G_{\text{xc}}^{\text{NN}}(\mathbf{g}).$$

Here, $G_{\text{xc}}^{\text{NN}}$ is the model's output, which is constructed using a four-layer fully-connected neural network. The descriptors defined in Eq. (12) include a set of five quantities, which form a *near region approximation* (NRA) in DFT. Depending on which terms are included, the formulation unifies various levels of detail about the local or quasi-local electron density. If all five descriptors are included, the ML-DFT model is referred to as NRA-type functional. To compute the XC potential from the XC energy density, a Monte Carlo method was used instead of backpropagation, avoiding complications from both the backpropagation through the inverse KS problem and the second-order derivative problem.

The resulting ML XC energy density model with local density descriptors shows a reasonable performance (see Fig. 4 in Ref. 24). However, the performance only becomes comparable to traditional hybrid functionals when the coarse-grained quasi-local density is included through the fifth descriptor (the NRA shown in the original paper). CCSD(T) and G2⁹⁵ results are used as the reference data.

C. XC fragment energy model

The HEDT guarantees the representability of the XC potential and XC energy (or energy density) by the quasi-local density. This one-to-one mapping between the local XC potential and quasi-local electron density can be utilized in several different ways. A slightly different approach from previous models is to divide the XC energy into contributions from naturally meaningful fractions (e.g., *atoms*).

As shown in Fig. 6, the electron density of a system is divided into four fragments, each with a unique mapping to the system's properties. When the mapping $\rho_{\text{frag},i} \mapsto E_i$ for any $i \in \{1, 2, 3, 4\}$ for each fragment's XC energy contribution $E_{\text{xc}} = \sum_i E_i$ is specified, it

uniquely determines a quasi-local XC functional $E_i = E_{\text{xc}}[\rho_{\text{frag},i}]$. This mapping is relatively straightforward to find with atomic division. The total XC energy of a molecule can be equated to the summation of XC energy contributions from constituent atoms, and a machine learning model can read and interpret quasi-local densities around each nucleus to output the corresponding atomic XC energy contribution.

It should be noted that even though the XC energy can be expressed as the summary of the contribution from individual atoms, even higher-order interactions among two or more atoms can still be partitioned into the single-atom contribution because the quasi-local density around each nucleus contains information from all orders. However, it is the machine learning model's capability to determine how the energy contribution is split among the participating atoms. For instance, for a C=O bond in a specific environment, the XC energy correction attributable to the bond can be apportioned to both the carbon and oxygen atoms.

Atomic contributions to molecular potential energy surfaces (PES) have been constructed prior to the widespread use of deep learning models, as demonstrated in the work of Behler and Parrinello.⁴⁹ However, to construct a truly universal XC functional that requires no additional information beyond the density itself, higher complexity models are necessary. Every aspect of the XC energy or potential arises from the subtle variations in the shape of the quasi-local density. Recent advancements in deep learning have made it possible to construct such models.

Dick and Fernandez-Serra¹⁶ successfully demonstrated promising accuracy in small molecules using an XC energy fragment model based on atomic contributions. The model constructs specific neural networks for each atom type and samples the electron density surrounding each nucleus using Gaussian-orbital-like projectors. Symmetrized projected values serve as the input for the neural networks, with the output representing energy contribution from each atom. The total XC energy is calculated by summing the outputs of all atomic neural networks. Functional derivatives are needed with respect to density for SCF calculation. Figure 7 depicts the architecture of the ML-DFT model. Noticeably, the derivatives assume

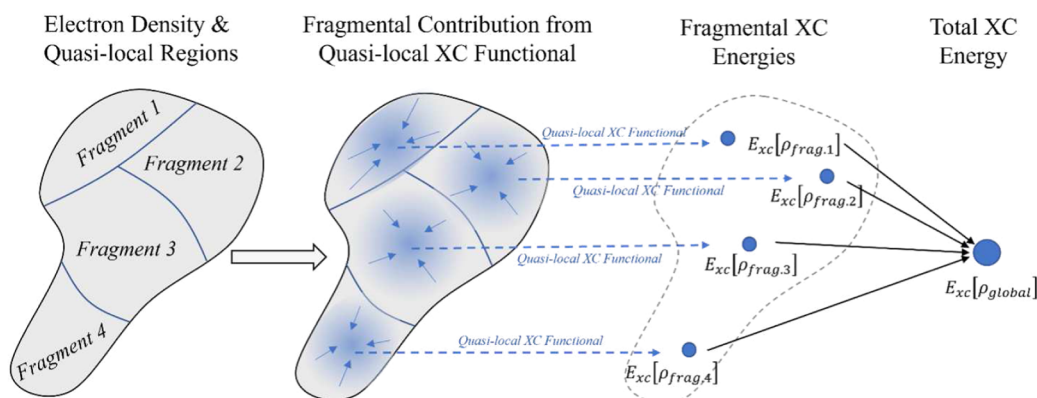


FIG. 6. The concept of the XC energy by fragments. In practice, the fragments are usually chemically meaningful parts like electron density around each nucleus (atom) in a molecule. From Wu *et al.*, *Quantum Chemistry in the Age of Machine Learning* (pp. 531–558). Copyright (2023) Elsevier. Reprinted with permission Elsevier.

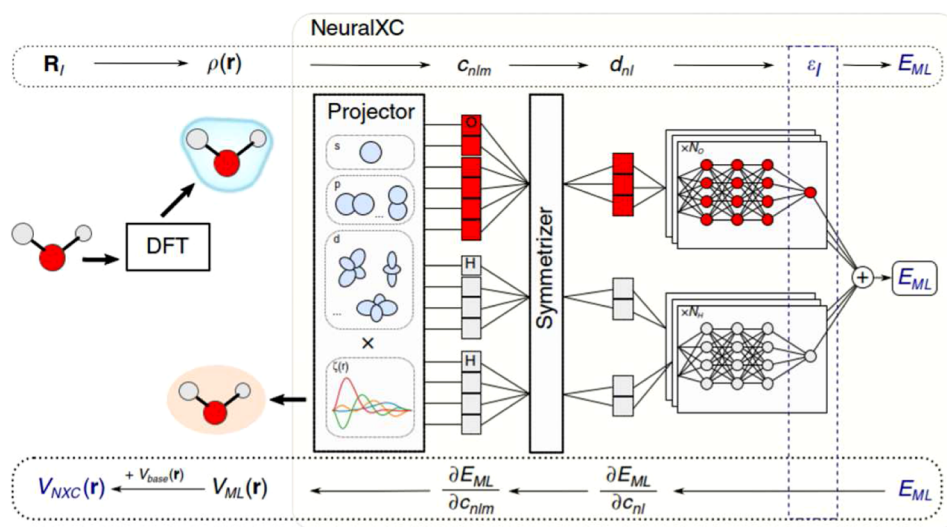


FIG. 7. The structure of the ML XC fragment energy model. Quasi-local electron density around each nucleus is described by *projectors*. Descriptors are then symmetrized and fed to the neural networks. Atoms of the same type share the same neural network parameters. Their respective outputs play the role of fragmental contributions, which are summed up to produce the total XC energy. Reproduced with permission from Dick and Fernandez-Serra, Nat. Commun. 11, 3509 (2020) Author(s), licensed under a Creative Commons Attribution 4.0 International License: <http://creativecommons.org/licenses/by/4.0/>.¹⁶

a rather simple transformation from density descriptors to density itself,

$$v_{\text{ML}}[\rho(\mathbf{r})] = \sum_{\beta} \frac{\partial E_{\text{ML}}}{\partial c_{\beta}} \frac{\delta c_{\beta}[\rho]}{\delta \rho(\mathbf{r})} = \sum_{\beta} \frac{\partial E_{\text{ML}}}{\partial c_{\beta}} \psi_{\beta}(\mathbf{r}), \quad (13)$$

where β is the index for different projectors, c_{β} is the projected value of the density of the projector, and $\psi_{\beta}(\mathbf{r})$ is the shape of the projector.

While the model developed by Dick *et al.* has shown promising accuracy for small molecules, it is not yet universal. The model relies on different neural networks and projectors for each type of atom, and different models were trained for different datasets. Specifically, the researchers developed three distinct models for three different datasets.

VII. IMPROVING ML DFT MODEL PERFORMANCE

In this section, we will review existing approaches to improve the performance of ML DFT models. These include the use of different training strategies, designing specific loss functions, and imposing physical constraints that density functionals should satisfy. By implementing these methods, we can improve the accuracy of DFT calculations and enhance.

A. Fully differentiable training with SCF calculations

To build a ML-DFT model that accurately represents the universal XC functional, the trained model with a fixed set of parameters should be applicable to any atoms, molecules, and materials. However, optimizing parameters during the training phase can be highly complicated due to the tangled relationship between the ML model and the SCF calculations. The parameters in the model should be

optimized in a way that aids the SCF procedure in converging to the correct density. If the same model is invoked during each SCF calculation, one may isolate the SCF procedure from the model training. This problem has been solved by implementing the KS equation with differential programming,^{96–98} which is an emerging programming paradigm allowing one to take the derivative of an output of an arbitrary code snippet with respect to its input using automatic differentiation techniques.⁶⁶

One can combine the SCF calculation within the optimization procedure to better train a ML-DFT model. This idea has been first demonstrated in a simple 1D system by Li *et al.*²¹ Later on, Kasim and Vinko¹⁹ and Dick and Fernandez-Serra⁹⁹ also implemented a neural network model for the three-dimensional molecules, where the derivatives can be computed by backpropagating through the SCF iterations. However, this approach requires a large amount of memory and may result in numerical instability when computing the derivatives, which makes it difficult to train on large dataset. One can apply the technique of implicit differentiation⁶⁹ to reduce the computational complexity and memory footprint of the actual implementation.

B. Designing loss functions

In supervised learning, the ML-DFT model is optimized by minimizing loss functions defined by the difference between the output values and those reference data. To train the model, it is common to use the following electron density loss:

$$L_{\rho} = \mathbb{E}_{\text{train}} \left[\int |\rho_{\text{ML-KS}}(\mathbf{r}) - \rho_{\text{target}}(\mathbf{r})|^2 d\mathbf{r} \right],$$

where $\rho_{\text{ML-KS}}$ is the electron density after KS-SCF calculation with the ML-DFT model for the XC functional or potential, ρ_{target} is

the target or reference electron density, and $\mathbb{E}_{\text{train}}[\cdot]$ indicates the averaging operation over the training set.

If the ML-DFT model is constructed to output the XC potential, one may skip solving the KS equations during training and impose the loss function in XC potential as

$$L_V = \mathbb{E}_{\text{train}} \left[\int |v_{\text{xc}}^{\text{ML}}(\mathbf{r}) - v_{\text{xc}}^{\text{target}}(\mathbf{r})|^2 d\mathbf{r} \right],$$

where $v_{\text{xc}}^{\text{ML}}$ is the ML XC potential, and $v_{\text{xc}}^{\text{target}}$ is the target or reference XC potential. In this case, the target potential should be pre-computed in the data preparation phase, and the model does not involve any SCF calculation during training. If the model output is the XC energy E_{xc} (or the XC energy density ϵ_{xc}), in addition to reproducing the electron density, the loss function in energy,

$$L_E = \mathbb{E}_{\text{train}} [E_{\text{xc}}^{\text{ML}} - E_{\text{xc}}^{\text{target}}]^2,$$

can also be added. This could be combined with other loss functions weighted by some hyper-parameters.

To construct an accurate ML-DFT model, it is important that the ML-DFT model not only reproduces the target energy but also reproduces the target electron density. The target electron density is often obtained from expensive *ab initio* methods. Gradient descent, or its variants, is commonly used for optimization during training. Automatic differentiation during backpropagation allows for effective computation of the gradient with respect to model parameters. If the density loss is included and the model is coupled with KS equations, backpropagation requires the inverse eigenvalue problem in the KS equations to be solved before parameter updates. This requires numerical techniques to access the network for parameter updates. Alternatively, reproducing the target density can be enforced by using the potential loss only as shown in Ref. 14.

C. Physical constraints for ML-DFT models

Although ML techniques have been widely employed for finding the exact form of universal XC functionals, these ML XC functionals are seen as *black boxes* and may not satisfy the physical constraints that the XC functional should obey in principle. For instance, the exchange-energy density of any finite many-electron system satisfies the exact $1/r$ asymptotic behavior.⁵⁸ This theoretical insight may be useful when designing parameterizations of new ML XC functionals. Moreover, other physical constraints, such as spin-scaling¹⁰⁰ for the exchange energy and the Lieb–Oxford bound¹⁰¹ for the exchange–correlation energy, are derived from fundamental principles of DFT and, thus, can also be used to guide the ML modeling.

Recent efforts of designing ML XC functionals satisfying certain physical constraints have been made to address this issue by integrating ML modeling and exact-constraint satisfaction.^{102,103} This approach has shown promise in producing ML constructed XC functionals that satisfy physical constraints and exhibit improved transferability and accuracy over traditional approximations.

VIII. OUTLOOK

A. General quasi-local descriptor formalism

The quasi-local electron density, which contains enough intrinsic information about the molecular system as dictated by the HEDT, is clearly a more suitable descriptor for training a better ML-DFT model compared to that of using either the local electron density or the global one. With the quasi-local electron density descriptors, one can parameterize the mapping from electron density to XC quantity with sufficiently many features to capture the details of the mapping. Once the electron density is given, the XC quantities are uniquely determined.

The general workflow of a quasi-local ML-DFT model is depicted in Fig. 8. The quasi-local electron density distribution $\rho_{\text{in}}(\mathbf{r}; \mathbf{r}_0)$ around \mathbf{r}_0 is inputted as the descriptors to the ML-DFT model; it outputs the intermediate XC potential $v_{\text{xc}}(\mathbf{r}_0)$ or XC energy density $\epsilon_{\text{xc}}(\mathbf{r}_0)$ at \mathbf{r}_0 that can be used in the subsequent KS solver. The input electron density function may be obtained by CCSD, the quantum Monte Carlo method, or other high-precision quantum chemistry methods. After the KS solver, a new charge density function ρ_{new} and other physical properties such as the total energy are obtained, and these can be used to form the loss function to train the ML model by comparing with the high-precision electron density and/or other quantities such as high-precision energy. Once the training is complete, the resulting ML-DFT model can be employed in the SCF calculation to calculate highly accurate physical properties such as electron density and total energy of the system.

Alternatively, ρ_{in} can be calculated via the conventional DFT methods, such as B3LYP, as the B3LYP version of ρ_{in} has a one-to-one correspondence to the higher-precision ρ_{in} (for instance, CCSD). The advantage is that no SCF calculation is required to calculate the molecular properties, once the ML-DFT model is built, and the input ρ_{in} can be obtained by employing the conventional DFT calculation.

One may extend the NN-based B3LYP functional developed in Ref. 5 into a quasi-local descriptor-based ML-DFT model. In this case, the ML-DFT model outputs a set of space-dependent coefficients $\{a_0, a_X, a_C\}$, which calibrates the original B3LYP functional. We remark that an additional correction term ΔE of the energy functional can also be added to enhance the model's capability to calculate the absolute energy. Besides the approach of outputting the space-dependent coefficients and the correction term, one may also directly target an energy density¹⁰⁴ of the underlying energy functional as an intermediate output. Adding energy density within the ML-DFT model might be useful to obtain either the XC potential

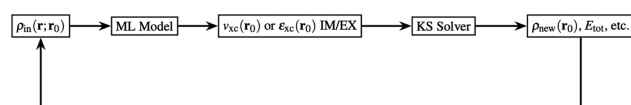


FIG. 8. A general quasi-local ML-DFT modeling workflow. The ML-DFT model takes the quasi-local electron density (or other descriptors) $\rho_{\text{in}}(\mathbf{r}; \mathbf{r}_0)$ and outputs an intermediate XC quantity (either v_{xc} or ϵ_{xc}); IM/EX simply means whether there are labels of the targeted quantity in the training steps. Then, the produced XC quantity is used in the KS solve to generate a new electron density ρ_{new} and thus the total energy E_{tot} (or any other quantities of interest).

(by automatic differentiation) or the XC functional (by numerical integration¹⁰⁵).

Recently, a new work based on the quasi-local electron density formulation of the ML-DFT model was published.¹⁰⁶ This is a quasi-local version of the electron density formulation of the NN-based ML-DFT model reported in Ref. 5. Instead of learning the mapping $\rho_{\text{quasi-local}} \rightarrow v_{\text{XC}}$ from scratch, the model learns the space-dependent coefficients combining three existing functionals as follows:

$$E_{\text{xc}}^{\text{MLP}}[\rho] = \int d\mathbf{r} \mathbf{f}_\theta(\mathbf{r}) \cdot \begin{bmatrix} \varepsilon_X^{\text{LDA}}(\mathbf{r}) \\ \varepsilon_X^{\text{HF}}(\mathbf{r}) \\ \varepsilon_X^{\omega\text{HF}}(\mathbf{r}) \end{bmatrix}. \quad (14)$$

In Eq. (14), \mathbf{f}_θ is a row vector of 3 elements outputted by the machine learning model, while $\varepsilon_X^{\text{LDA}}(\mathbf{r})$, $\varepsilon_X^{\text{HF}}(\mathbf{r})$, and $\varepsilon_X^{\omega\text{HF}}(\mathbf{r})$ are the local LDA,⁵⁶ local Hartree–Fock, and local range-separated Hartree–Fock energy densities (see Ref. 107), respectively. An extra D3¹⁰⁸ correction was added to the ML functional $E_{\text{xc}}^{\text{MLP}}$ to produce the final XC energy prediction.

For the fragment energy model, the density-weighted interaction in Eq. (11) can be replaced by a summation over fragmental contribution,

$$v_{\text{xc}}(\mathbf{r}) = \frac{\delta E_{\text{xc}}}{\delta \rho(\mathbf{r})} = \sum_{i: \mathbf{r} \in \text{frag}, i} \frac{\delta E_i}{\delta \rho(\mathbf{r})}.$$

As the boundaries between fragments are not always clear-cut [for example in Eq. (13), the projector c_β has a kernel of Gaussian orbital shape¹⁶], there can be multiple fragment energies contributing to the potential at any given positions, allowing for a smooth transition between fragments.

Once the ML-DFT model is trained for a specific type of XC quantities, it can be incorporated into the SCF calculations and subsequently used for post-processing the molecular properties of interest, as in traditional DFT calculations. The quasi-local density descriptor approach emerges as the mainstream approach to constructing the ML-DFT model. The remaining is how best to design and represent the quasi-local electron density. Moreover, the electron density is also being used as the target, as it is the key entity in DFT and contains the health of information. More research is expected in this direction.

B. ML models for van der Waals interaction

An accurate description of the van der Waals (vdW) interaction is challenging for traditional DFT, as it is weak and is due to the interaction of transient atomic dipoles. While some conventional DFT approximations have shown remarkable performance in certain systems,¹⁰⁹ they often rely on nonlocal quantities that make them difficult to apply in the quasi-local ML-DFT method.

Because the vdW interaction is caused by the interaction among transient atomic dipole moments, it can be, in principle, machine-learned from the electron density. As the vdW is weak, a minute change in electron density is induced. The minor changes in density and their corresponding XC potential are both higher order effects in a perturbative sense rather than the cause of vdW interaction. It is thus possible to machine-learn the vdW interaction directly from the electron density ignoring the high-order electron density changes.

Recall that the XC energy and potential can be written as follows: for a given ρ_0 with a small perturbation $\delta\rho$,

$$\begin{aligned} E_{\text{xc}}[\rho_0 + \delta\rho] &\approx E_{\text{xc}}[\rho_0] + \int d\mathbf{r} \left. \frac{\delta E_{\text{xc}}[\rho]}{\delta \rho(\mathbf{r})} \right|_{\rho=\rho_0} \delta\rho(\mathbf{r}), \\ v_{\text{xc}}[\rho_0 + \delta\rho](\mathbf{r}) &= \left. \frac{\delta E_{\text{xc}}[\rho]}{\delta \rho(\mathbf{r})} \right|_{\rho=\rho_0+\delta\rho} \approx \left. \frac{\delta E_{\text{xc}}[\rho]}{\delta \rho(\mathbf{r})} \right|_{\rho=\rho_0} \\ &\quad + \int_{\mathbf{r}' \in B(\mathbf{r})} d\mathbf{r}' \frac{\delta}{\delta \rho(\mathbf{r}')} \left[\left. \frac{\delta E_{\text{xc}}[\rho]}{\delta \rho(\mathbf{r})} \right] \right|_{\rho=\rho_0} \delta\rho(\mathbf{r}'). \end{aligned} \quad (15)$$

It is evident from Eq. (15) that the minor density change resulting from the vdW interaction can be mostly ignored when calculating the XC potential during SCF, within reasonable accuracy requirements. The second term, which accounts for second-order variation in the electron density, is significantly smaller than the first term, as the change in density for vdW interaction is minimal. However, the energy shift due to vdW interaction is significant and cannot be neglected. Therefore, including an additional correction term for vdW interaction after SCF calculation is a reasonable approach. A separate vdW ML model can be trained using the quasi-local electron density and added to the current ML XC model as an extra correction term to the XC energy.

Empirical correction approaches, like the widely-used DFT-D3 method,¹⁰⁸ are computationally efficient but limited in their effectiveness due to their reliance on a few empirical parameters and their sensitivity to specific systems. On the other hand, a customized ML model with a large number of tunable parameters and degrees of freedom may bring significant improvements.

Recently, Proppe *et al.* employed Gaussian process regression¹¹⁰ to correct systematic errors in DFT calculation with D3-type dispersion corrections.¹¹¹ This model is referred to as D3-GP in the original work. The training data, consisting of 1248 samples of molecular dimers, are the differences between interaction energies obtained from PBE-D3(BJ)^{108,112,113}/ma-def2-QZVPP^{114,115} and DLPNO-CCSD(T)^{116,117}/CBS¹¹⁸ calculations. Once provided with reference data for new molecular systems, the underlying D3-GP model can learn to adapt to these and similar systems. The D3-GP model outperforms the existing PBE-specific correction schemes^{113,119,120} with respect to three different validation sets. One may expect that with sufficient training data, an ML model for vdW correction is likely to outperform existing empirical models for dispersion correction. Once the ML-vdW model is trained and validated, combining this ML model with the quasi-local ML-DFT model is straightforward.

C. Other future research directions

The full potential of the ML-DFT model can be explored by utilizing larger and more diverse datasets that can significantly benefit the modeling of ML-DFT calculations. By incorporating diverse molecules, chemical environments, and properties, the ML-DFT models can capture finer details of the exchange-correlation interaction and thus improve the model's generalizability. Expanding the dataset to include molecules with various sizes, complexities, and properties would enhance the training of ML-DFT models and enable more accurate representations for the XC quantities. While maintaining the efficiency of model training would become

challenging, larger models with a higher number of parameters may effectively capture intricate features and correlations in the data, leading to improved accuracy and reliability in ML-DFT models.

Recently, the notion of neural operator¹²¹ and the technique of operator learning¹²² gain much attention from different scientific communities. The goal of operator learning is to seek a directly functional relation that maps elements from an infinite dimensional space to another infinite dimensional one. One of the great features of operator learning is that the parameterization of the mapping is discretization invariance, i.e., the resulting mapping from ML models is independent of the resolution of input and output data, as the operator learning model aims to learn the intrinsic structure of the map between the abstract spaces. One may expect that this approach could benefit the explore of XC functionals that map electron densities, which are smooth functions of spatial variables, to the energies of the underlying quantum systems. Moreover, by incorporating domain knowledge and physical constraints, ML-DFT models may have better representability for the exchange-correlation quantities, leading to the development of more accurate and physically meaningful XC functionals.

IX. CONCLUDING REMARKS

The explosive development in AI has catalyzed a quick turnover of machine-learning models for density functional theory. From an algorithmic perspective, most of the above-mentioned approaches have focused on applying ML architectures such as artificial or convolutional neural networks to learn the XC functionals. However, other promising candidates, such as graph neural networks (GNN), recurrent neural networks (RNN),¹²³ and transformers,¹² are also being explored for overhauling the design of XC functionals. GNN extends CNN toward irregular grids for electron density or XC potential. RNN is ideal for time-dependent data and may find profound applications in time-dependent DFT. On the other hand, transformers and other attention-based models allow the model to be smarter by deciding where to pay attention in the electron density or XC potential. Given the subtlety and sensitivity of electron density data in DFT problems, attention-based models may be a good fit.

Here, we have reviewed the machine learning approaches for constructing XC-related quantities (such as energy functional or potential) in DFT. The review began with a discussion of two pioneering works, ML-DFT models that use global descriptors and progressed toward more intuitive and transferable quasi-local models, concluding with an additional ML term for vdW interaction. For the quasi-local descriptor models, we introduced the holographic electron density theorem as the theoretical foundation and presented a series of successful implementation schemes. All quasi-local ML-DFT models (such as the ML XC potential model) share the same fundamental design elements and have deep physical connections. We have demonstrated successful stories for these variants,^{14,16,21,24} and we encourage readers to read the respective original papers, as well as the open-source codes and examples provided. We hope that new generations of ML-DFT models will accurately construct the universal XC functional of DFT in the near future, revolutionizing the field of quantum chemistry, similar to how AlphaFold¹²⁴ has transformed the field of structural biology.

Forward looking, the eventual ML-DFT model for the XC functional should have the following features. First, the descriptors should be made of the quasi-local electron density. Second, the targets should include the high precision electron density; and this can be the explicit target, or the implicit, for instance, in Ref. 5, the explicit target is the XC potential, which, in turn, leads to the high precision electron density by solving the KS equation. Finally, the XC potential and energy density can be the output or the intermediate target that leads to the target electron density. An additional machine-learning module for the vdW interaction may also be included in the workflow to deal with the weak interaction of transient atomic dipoles. Ultimately, the ML-DFT model combined with the vdW interaction module should be able to accurately reproduce the target energy and the target electron density for any molecular system.

AUTHOR DECLARATIONS

Conflict of Interest

The authors have no conflicts to disclose.

Author Contributions

Jiang Wu: Conceptualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Sai-Mang Pun:** Conceptualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **Xiao Zheng:** Conceptualization (equal); Writing – original draft (equal); Writing – review & editing (equal). **GuanHua Chen:** Conceptualization (lead); Writing – original draft (equal); Writing – review & editing (equal).

DATA AVAILABILITY

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

APPENDIX A: TUTORIAL OF THE ML-DFT XC POTENTIAL MODEL

For a hands-on experience, the readers are encouraged to try out our open-source package on GitHub.¹²⁵ Most of the codes were written in Python, and the ML models were built with the open-source package PyTorch.¹²⁶ For a better understanding of the implementation details, inexperienced readers are recommended to go through a comprehensive tutorial of PyTorch, before making any modifications to the models we provide. As a starting point, PyTorch provides tutorials of introductory level on their own website.

As a simple example, we make use of our model here using the XC potential generated by the Wu–Yang (WY) method⁹¹ as the direct training set. For H₂ molecule (see Fig. 9 for the numerical results), training can be performed with the pre-calculated XC potential for H₂ for various H–H bond distances, and no SCF calculation is needed. At the evaluation phase, full SCF calculation can be performed for each structure, and it is implemented with the PySCF package.¹²⁷ To perform training and

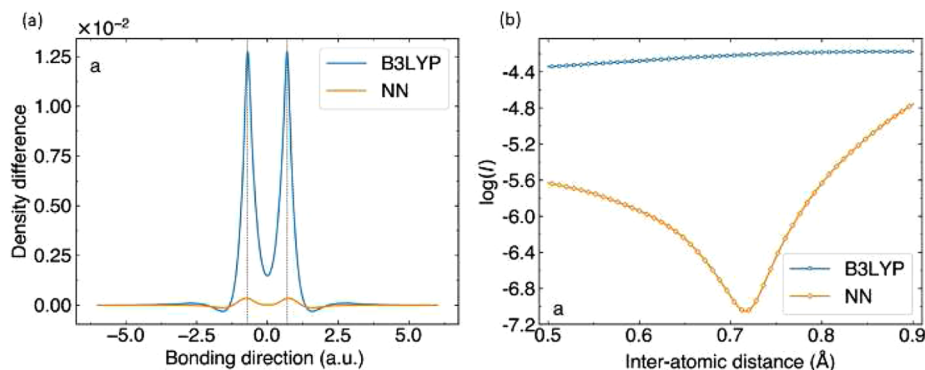


FIG. 9. (a) A typical (with a reasonable bond distance) SCF run for the trained model of the H_2 example will produce density errors comparable to NN. (b) The I value will be comparable to the corresponding value on the NN curve, which is significantly lower than the error of B3LYP. From Wu *et al.*, *Quantum Chemistry in the Age of Machine Learning* (pp. 531–558). Copyright (2023) Elsevier. Reprinted with permission Elsevier.

evaluation on this example, one may walk through the following steps below:

1. Before getting started, make sure all the prerequisites are installed and work properly.
2. Create and enter a new folder; download the code and dataset by typing

```
$ git clone https://github.com/zhouyyc6782/oe-p-wy-xcnn.git
```

3. Enter the `example/simple_H2` directory. Create a folder called `log` by typing

```
$ mkdir log
to store the upcoming results and logged files.
```

4. Start training by typing

```
$ python ../nn_train/main.py train.cfg
```

Here, all training settings and hyper-parameters are defined in the `.cfg` file; to write a new `.cfg` file for a different configuration, please refer to the README file provided with the GitHub repo.

5. Training will start on the provided H_2 dataset; by default, the number of epochs is 1000.

6. Perform SCF calculations on the newly trained model by typing

```
$ python ../xcnn/main.py test.cfg
```

7. One can check the SCF performance of the model by examining the output file generated. A typical run for a small molecule like H_2 should result in an error at the level of 10^{-5} – 10^{-7} in terms of I value. Since only one H_2 structure is included in the `simple_H2` training set, the error could be larger. Here, the I value between two (possibly different) densities is defined to be

$$I = I[\rho_1, \rho_2] = \frac{\int |\rho_1(\mathbf{r}) - \rho_2(\mathbf{r})|^2 d\mathbf{r}}{\int |\rho_1(\mathbf{r})|^2 d\mathbf{r} + \int |\rho_2(\mathbf{r})|^2 d\mathbf{r}}$$

This tutorial is centered on a pre-built dataset from one H_2 structure for both training and SCF. To reproduce the result from

the original paper,¹⁴ a modified and re-compiled version of PySCF is needed for generating WY target data from scratch with the codes in the folder `oe-p-wy` (while the SCF part only needs the vanilla version of PySCF). One can refer to the README in the GitHub repo for more details on installing a custom version of the PySCF package.

Those scripts (e.g., `run_oe.py`, `gen_dataset.py`, `run_train.py`, and `run_xcnn.py`) in the repo provide automatic scripts for generating data from WY calculation, collecting data, training the model with the data, and testing the model with SCF procedure, respectively. Interested readers are advised to follow the README from the GitHub repository in step 2) for re-compiling PySCF and additional custom implementations of the codes.

The codes provided within this tutorial constitute (i) data generation (with the WY method), (ii) training part, and (iii) SCF computation. One can build their own codes based on this GitHub repo for molecules or ions other than H_2 in this simple example. Depending on the format of the dataset, one needs to write their own scripts like `run_oe.py`, `gen_dataset.py`, `run_train.py`, and `run_xcnn.py`, for automating the whole algorithmic procedure.

APPENDIX B: OPTIMIZED EFFECTIVE POTENTIAL AND DATA GENERATION

The electron densities that are employed to train the ML models can be obtained using highly accurate *ab initio* methods such as wave-function based methods like CCSD.⁷² Besides the electron density, the values of XC potential are also needed. Given a density computed from CCSD, the corresponding XC potential can be calculated by various optimization procedures that effectively invert the KS equations (collectively referred to as the inverse Kohn–Sham methods; see also Ref. 128). The optimization procedure employed in Ref. 14 to generate a training dataset is the so-called Wu–Yang method (WY) developed in Ref. 91, which will be briefly elaborated here.

Readers might wonder that if a numerical optimization procedure can resolve XC potential from electron density, then why do we bother training an ML model that does the exact same thing? The answer lies in the core concept of DFT itself. What we want to predict from the ML model is the universal XC functional that maps

any density to its corresponding XC potential. On the other hand, the optimization procedure only solves system-specific XC potential that is associated with a particular known electron density. The procedure entails only the mathematics of inverting KS equations, which does not include the physics of the many-particle system at all. In contrast, the ML model tries to learn the intrinsic physics behind it, which is by definition fundamental. Those values of electron densities and XC potentials generated by inverse KS methods are fed to the ML model as training data.

The solution to the inverse KS problem is not as straightforward as it first appears. An analytical solution is mostly absent and different kinds of numerical optimization techniques are usually employed. One of the popular potential optimization schemes was invented by Wu and Yang in Refs. 91 and 92. For a given input density ρ_{in} , one first constructs a Lagrangian, denoted as W_s , in terms of the total effective potential (denoted as v) and the single particle wave functions (denoted as ϕ_i 's),

$$W_s[\Psi(v(\mathbf{r})), v(\mathbf{r})] = 2 \sum_{i=1}^{\text{occ}} \langle \phi_i | \hat{T} | \phi_i \rangle + \int d\mathbf{r} v(\mathbf{r}) (\rho(\mathbf{r}) - \rho_{\text{in}}(\mathbf{r})), \quad (\text{B1})$$

where $\Psi = (N!)^{-1/2} \det(\phi_i(x_j))$ is Slater's determinant function associated with the orbitals ϕ_i 's, and $v(\mathbf{r})$ serves as a Lagrange multiplier. When W_s is stationary with respect to v , the electron density becomes the same as the given density ρ_{in} and

$$\frac{\delta W_s[\Psi(v(\mathbf{r})), v(\mathbf{r})]}{\delta v(\mathbf{r})} = \rho(\mathbf{r}) - \rho_{\text{in}}(\mathbf{r}) = 0. \quad (\text{B2})$$

In practice, the potential is projected onto a set of Gaussian basis functions.¹²⁹ Once the effective potential is calculated, the XC potential v_{xc} can be easily found by subtracting the external and the Hartree potentials.¹³⁰

With the pair of density and XC potential being obtained, the training procedure is decoupled from the KS SCF procedure, and the resulting ML model converts its inputs ρ into the outputs v_{xc} . Training proceeds with a typical backpropagation procedure with an optimizer using stochastic gradient descent (SGD)¹³¹ or the Adam method.¹³² Once large enough data are accessible for various types of molecules and quasi-local environments, the parameters in the ML XC potential model can be better trained and yield a more accurate and universal XC potential of real molecular systems.

REFERENCES

- 1 P. Hohenberg and W. Kohn, "Inhomogeneous electron gas," *Phys. Rev.* **136**, B864 (1964).
- 2 W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects," *Phys. Rev.* **140**, A1133 (1965).
- 3 A. M. Teale, T. Helgaker, A. Savin, C. Adamo, B. Aradi, A. V. Arbuznikov, P. W. Ayers, E. J. Baerends, V. Barone, and P. Calaminici, "DFT exchange: Sharing perspectives on the workhorse of quantum chemistry and materials science," *Phys. Chem. Chem. Phys.* **24**, 28700–28781 (2022).
- 4 D. J. Tozer, V. E. Ingamells, and N. C. Handy, "Exchange-correlation potentials," *J. Chem. Phys.* **105**, 9200–9213 (1996).
- 5 X. Zheng, L. Hu, X. Wang, and G. Chen, "A generalized exchange-correlation functional: The neural-networks approach," *Chem. Phys. Lett.* **390**, 186–192 (2004).
- 6 G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Z. Li, and T. Hospedales, "When face recognition meets with deep learning: An evaluation of convolutional neural networks for face recognition," in *Proceedings of the IEEE International Conference on Computer Vision Workshops* (IEEE, 2015), pp. 142–150.
- 7 T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *IEEE Comput. Intell. Mag.* **13**, 55–75 (2018).
- 8 A. W. Senior, R. Evans, J. Jumper, J. Kirkpatrick, L. Sifre, T. Green, C. Qin, A. Židek, A. W. R. Nelson, and A. Bridgland, "Improved protein structure prediction using potentials from deep learning," *Nature* **577**, 706–710 (2020).
- 9 D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, and M. Lanctot, "Mastering the game of Go with deep neural networks and tree search," *Nature* **529**, 484–489 (2016).
- 10 Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural Comput.* **1**, 541–551 (1989).
- 11 M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proceedings of 2005 IEEE International Joint Conference on Neural Networks* (IEEE, 2005), Vol. 2, pp. 729–734.
- 12 A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems* [Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2017], Vol. 30.
- 13 R. Nagai, R. Akashi, S. Sasaki, and S. Tsuneyuki, "Neural-network Kohn-Sham exchange-correlation potential and its out-of-training transferability," *J. Chem. Phys.* **148**, 241737 (2018).
- 14 Y. Zhou, J. Wu, S. Chen, and G. Chen, "Toward the exact exchange-correlation potential: A three-dimensional convolutional neural network construct," *J. Phys. Chem. Lett.* **10**, 7264–7269 (2019).
- 15 E. Cuierrier, P.-O. Roy, and M. Ernzerhof, "Constructing and representing exchange-correlation holes through artificial neural networks," *J. Chem. Phys.* **155**, 174121 (2021).
- 16 S. Dick and M. Fernandez-Serra, "Machine learning accurate exchange and correlation functionals of the electronic density," *Nat. Commun.* **11**, 3509 (2020).
- 17 R. Han, M. Rodríguez-Mayorga, and S. Lubner, "A machine learning approach for MP2 correlation energies and its application to organic compounds," *J. Chem. Theory Comput.* **17**, 777–790 (2021).
- 18 Y. Ikabata, R. Fujisawa, J. Seino, T. Yoshikawa, and H. Nakai, "Machine-learned electron correlation model based on frozen core approximation," *J. Chem. Phys.* **153**, 184108 (2020).
- 19 M. F. Kasim and S. M. Vinko, "Learning the exchange-correlation functional from nature with fully differentiable density functional theory," *Phys. Rev. Lett.* **127**, 126403 (2021).
- 20 X. Lei and A. J. Medford, "Design and analysis of machine learning exchange-correlation functionals via rotationally invariant convolutional descriptors," *Phys. Rev. Mater.* **3**, 063801 (2019).
- 21 L. Li, S. Hoyer, R. Pederson, R. Sun, E. D. Cubuk, P. Riley, K. Burke *et al.*, "Kohn-Sham equations as regularizer: Building prior knowledge into machine-learned physics," *Phys. Rev. Lett.* **126**, 036401 (2021).
- 22 J. T. Margraf and K. Reuter, "Making the coupled cluster correlation energy machine-learnable," *J. Phys. Chem. A* **122**, 6343–6348 (2018).
- 23 J. T. Margraf and K. Reuter, "Pure non-local machine-learned density functional theory for electron correlation," *Nat. Commun.* **12**, 344 (2021).
- 24 R. Nagai, R. Akashi, and O. Sugino, "Completing density functional theory by machine learning hidden messages from molecules," *npj Comput. Mater.* **6**, 43 (2020).
- 25 T. Nudujima, Y. Ikabata, J. Seino, T. Yoshikawa, and H. Nakai, "Machine-learned electron correlation model based on correlation energy density at complete basis set limit," *J. Chem. Phys.* **151**, 024104 (2019).
- 26 J. Schmidt, C. L. Benavides-Riveros, and M. A. L. Marques, "Machine learning the physical nonlocal exchange-correlation functional of density-functional theory," *J. Phys. Chem. Lett.* **10**, 6425–6431 (2019).
- 27 J. Wang, D. Zhang, R.-X. Xu, C. Yam, G. Chen, and X. Zheng, "Improving density functional prediction of molecular thermochemical properties with a machine-learning-corrected generalized gradient approximation," *J. Phys. Chem. A* **126**, 970–978 (2022).
- 28 M. Bogojeski, L. Vogt-Maranto, M. E. Tuckerman, K.-R. Müller, and K. Burke, "Quantum chemical accuracy from density functional approximations via machine learning," *Nat. Commun.* **11**, 5223 (2020).

- ²⁹L. Cheng, M. Welborn, A. S. Christensen, and T. F. Miller III, "A universal density matrix functional from molecular orbital-based machine learning: Transferability across organic molecules," *J. Chem. Phys.* **150**, 131103 (2019).
- ³⁰X.-M. Duan, Z.-H. Li, G.-L. Song, W.-N. Wang, G.-H. Chen, and K.-N. Fan, "Neural network correction for heats of formation with a larger experimental training set and new descriptors," *Chem. Phys. Lett.* **410**, 125–130 (2005).
- ³¹L. Hu, X. Wang, L. Wong, and G. Chen, "Combined first-principles calculation and neural-network correction approach for heat of formation," *J. Chem. Phys.* **119**, 11501–11507 (2003).
- ³²H. Ji and Y. Jung, "A local environment descriptor for machine-learned density functional theory at the generalized gradient approximation level," *J. Chem. Phys.* **148**, 241742 (2018).
- ³³H. Li, L. Shi, M. Zhang, Z. Su, X. Wang, L. Hu, and G. Chen, "Improving the accuracy of density-functional theory calculation: The genetic algorithm and neural network approach," *J. Chem. Phys.* **126**, 144101 (2007).
- ³⁴Q. Liu, J. Wang, P. Du, L. Hu, X. Zheng, and G. Chen, "Improving the performance of long-range-corrected exchange-correlation functional with an embedded neural network," *J. Phys. Chem. A* **121**, 7273–7281 (2017).
- ³⁵K. Ryczko, D. A. Strubbe, and I. Tamblyn, "Deep learning and density-functional theory," *Phys. Rev. A* **100**, 022512 (2019).
- ³⁶J. Sun, J. Wu, T. Song, L. Hu, K. Shan, and G. Chen, "Alternative approach to chemical accuracy: A neural networks-based first-principles method for heat of formation of molecules made of H, C, N, O, F, S, and Cl," *J. Phys. Chem. A* **118**, 9120–9131 (2014).
- ³⁷M. Welborn, L. Cheng, and T. F. Miller III, "Transferability in machine learning for electronic structure via the molecular orbital basis," *J. Chem. Theory Comput.* **14**, 4772–4779 (2018).
- ³⁸J. Wu and X. Xu, "The X1 method for accurate and efficient prediction of heats of formation," *J. Chem. Phys.* **127**, 214105 (2007).
- ³⁹G. Yang, J. Wu, S. Chen, W. Zhou, J. Sun, and G. Chen, "Size-independent neural networks based first-principles method for accurate prediction of heat of formation of fuels," *J. Chem. Phys.* **148**, 241738 (2018).
- ⁴⁰K. Ryczko, S. J. Wetzel, R. G. Melko, and I. Tamblyn, "Toward orbital-free density functional theory with small data sets and deep learning," *J. Chem. Theory Comput.* **18**, 1122–1128 (2022).
- ⁴¹F. Brockherde, L. Vogt, L. Li, M. E. Tuckerman, K. Burke, and K.-R. Müller, "Bypassing the Kohn-Sham equations with machine learning," *Nat. Commun.* **8**, 872 (2017).
- ⁴²P. Golub and S. Manzhos, "Kinetic energy densities based on the fourth order gradient expansion: Performance in different classes of materials and improvement via machine learning," *Phys. Chem. Chem. Phys.* **21**, 378–395 (2019).
- ⁴³L. Li, J. C. Snyder, I. M. Pelaschier, J. Huang, U.-N. Niranjan, P. Duncan, M. Rupp, K.-R. Müller, and K. Burke, "Understanding machine-learned density functionals," *Int. J. Quantum Chem.* **116**, 819–833 (2016).
- ⁴⁴R. Meyer, M. Weichselbaum, and A. W. Hauser, "Machine learning approaches toward orbital-free density functional theory: Simultaneous training on the kinetic energy density functional and its functional derivative," *J. Chem. Theory Comput.* **16**, 5685–5694 (2020).
- ⁴⁵J. Seino, R. Kageyama, M. Fujinami, Y. Ikabata, and H. Nakai, "Semi-local machine-learned kinetic energy density functional with third-order gradients of electron density," *J. Chem. Phys.* **148**, 241705 (2018).
- ⁴⁶J. Seino, R. Kageyama, M. Fujinami, Y. Ikabata, and H. Nakai, "Semi-local machine-learned kinetic energy density functional demonstrating smooth potential energy curves," *Chem. Phys. Lett.* **734**, 136732 (2019).
- ⁴⁷J. C. Snyder, M. Rupp, K. Hansen, K.-R. Müller, and K. Burke, "Finding density functionals with machine learning," *Phys. Rev. Lett.* **108**, 253002 (2012).
- ⁴⁸K. Yao and J. Parkhill, "Kinetic energy of hydrocarbons as a function of electron density and convolutional neural networks," *J. Chem. Theory Comput.* **12**, 1139–1147 (2016).
- ⁴⁹J. Behler and M. Parrinello, "Generalized neural-network representation of high-dimensional potential-energy surfaces," *Phys. Rev. Lett.* **98**, 146401 (2007).
- ⁵⁰H. Ma, A. Narayanaswamy, P. Riley, and L. Li, "Evolving symbolic density functionals," *Sci. Adv.* **8**, eabq0279 (2022).
- ⁵¹M. Gastegger, L. González, and P. Marquetand, "Exploring density functional subspaces with genetic algorithms," *Monatsh. Chem.-Chem. Mon.* **150**, 173–182 (2019).
- ⁵²R. Nagai and R. Akashi, "Development of exchange-correlation functionals assisted by machine learning," *arXiv:2206.15370* (2022).
- ⁵³P. Echenique and J. L. Alonso, "A mathematical and computational review of Hartree-Fock SCF methods in quantum chemistry," *Mol. Phys.* **105**, 3057–3098 (2007).
- ⁵⁴Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature* **521**, 436–444 (2015).
- ⁵⁵C. Lee, W. Yang, and R. G. Parr, "Development of the Colle-Salvetti correlation-energy formula into a functional of the electron density," *Phys. Rev. B* **37**, 785 (1988).
- ⁵⁶J. C. Slater, "A simplification of the Hartree-Fock method," *Phys. Rev.* **81**, 385 (1951).
- ⁵⁷V. Fock, "Näherungsmethode zur Lösung des quantenmechanischen Mehrkörperproblems," *Z. Phys.* **61**, 126–148 (1930).
- ⁵⁸A. D. Becke, "Density-functional exchange-energy approximation with correct asymptotic behavior," *Phys. Rev. A* **38**, 3098 (1988).
- ⁵⁹S. H. Vosko, L. Wilk, and M. Nusair, "Accurate spin-dependent electron liquid correlation energies for local spin density calculations: A critical analysis," *Can. J. Phys.* **58**, 1200–1211 (1980).
- ⁶⁰R. A. Vargas-Hernández, "Bayesian optimization for calibrating and selecting hybrid-density functional models," *J. Phys. Chem. A* **124**, 4053–4061 (2020).
- ⁶¹R. Kobayashi, N. C. Handy, R. D. Amos, G. W. Trucks, M. J. Frisch, and J. A. Pople, "Gradient theory applied to the Brueckner doubles method," *J. Chem. Phys.* **95**, 6723–6733 (1991).
- ⁶²Q. Zhao, R. C. Morrison, and R. G. Parr, "From electron densities to Kohn-Sham kinetic energies, orbital energies, exchange-correlation potentials, and exchange-correlation energies," *Phys. Rev. A* **50**, 2138 (1994).
- ⁶³J. Wu, G. Chen, J. Wang, and X. Zheng, "Redesigning density functional theory with machine learning," in *Quantum Chemistry in the Age of Machine Learning* (Elsevier, 2023), pp. 531–558.
- ⁶⁴E. A. Nadaraya, "On estimating regression," *Theory Probab. Its Appl.* **9**, 141–142 (1964).
- ⁶⁵S. Raschka, "Model evaluation, model selection, and algorithm selection in machine learning," *arXiv:1811.12808* (2018).
- ⁶⁶C. C. Margossian, "A review of automatic differentiation and its efficient implementation," *Wiley Interdiscip. Rev.: Data Min. Knowl. Discovery* **9**, e1305 (2019).
- ⁶⁷T. Tamayo-Mendoza, C. Kreisbeck, R. Lindh, and A. Aspuru-Guzik, "Automatic differentiation in quantum chemistry with applications to fully variational Hartree-Fock," *ACS Cent. Sci.* **4**, 559–566 (2018).
- ⁶⁸N. Yoshikawa and M. Sumita, "Automatic differentiation for the direct minimization approach to the Hartree-Fock method," *J. Phys. Chem. A* **126**, 8487–8493 (2022).
- ⁶⁹X. Zhang and G. K.-L. Chan, "Differentiable quantum chemistry with PySCF for molecules and materials at the mean-field level and beyond," *J. Chem. Phys.* **157**, 204801 (2022).
- ⁷⁰A. S. Abbott, B. Z. Abbott, J. M. Turney, and H. F. Schaefer III, "Arbitrary-order derivatives of quantum chemical methods via automatic differentiation," *J. Phys. Chem. Lett.* **12**, 3232–3239 (2021).
- ⁷¹B. Kalita, R. Pederson, J. Chen, L. Li, and K. Burke, "How well does Kohn-Sham regularizer work for weakly correlated systems?," *J. Phys. Chem. Lett.* **13**, 2540–2547 (2022).
- ⁷²J. Čížek, "On the correlation problem in atomic and molecular systems. Calculation of wavefunction components in Ursell-type expansion using quantum-field theoretical methods," *J. Chem. Phys.* **45**, 4256–4266 (1966).
- ⁷³W. L. McMillan, "Ground state of liquid He⁴," *Phys. Rev.* **138**, A442 (1965).
- ⁷⁴N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculations by fast computing machines," *J. Chem. Phys.* **21**, 1087–1092 (1953).
- ⁷⁵J. Riess and W. Münch, "The theorem of Hohenberg and Kohn for subdomains of a quantum system," *Theor. Chim. Acta* **58**, 295–300 (1981).

- ⁷⁶S. Fournais, M. Hoffmann-Ostenhof, T. Hoffmann-Ostenhof, and T. Østergaard Sørensen, "Analyticity of the density of electronic wavefunctions," *Ark. Mat.* **42**, 87–106 (2004).
- ⁷⁷S. Fournais, M. Hoffmann-Ostenhof, T. Hoffmann-Ostenhof, and T. Ø. Sørensen, "The electron density is smooth away from the nuclei," [math-ph/0109020](https://arxiv.org/abs/math-ph/0109020) (2001).
- ⁷⁸T. Jecko, "A new proof of the analyticity of the electronic density of molecules," *Lett. Math. Phys.* **93**, 73–83 (2010).
- ⁷⁹P. G. Mezey, "The holographic electron density theorem and quantum similarity measures," *Mol. Phys.* **96**, 169–178 (1999).
- ⁸⁰R. Carbó-Dorca and E. Besalú, "Communications on quantum similarity (2): A geometric discussion on holographic electron density theorem and confined quantum similarity measures," *J. Comput. Chem.* **31**, 2452–2462 (2010).
- ⁸¹P. Geerlings, G. Boon, C. Van Alsenoy, and F. De Proft, "Density functional theory and quantum similarity," *Int. J. Quantum Chem.* **101**, 722–732 (2005).
- ⁸²S. G. Krantz and H. R. Parks, *A Primer of Real Analytic Functions* (Springer Science & Business Media, 2002).
- ⁸³X. Zheng, F. Wang, C. Y. Yam, Y. Mo, and G. Chen, "Time-dependent density-functional theory for open systems," *Phys. Rev. B* **75**, 195127 (2007).
- ⁸⁴X. Zheng and G. Chen, "First-principles method for open electronic systems," in *Nanoscale Phenomena: Basic Science to Device Applications* (Springer, 2008), pp. 235–243.
- ⁸⁵X. Zheng, G. Chen, Y. Mo, S. Koo, H. Tian, C. Yam, and Y. Yan, "Time-dependent density functional theory for quantum transport," *J. Chem. Phys.* **133**, 114101 (2010).
- ⁸⁶X. Zheng, C. Yam, F. Wang, and G. Chen, "Existence of time-dependent density-functional theory for open electronic systems: Time-dependent holographic electron density theorem," *Phys. Chem. Chem. Phys.* **13**, 14358–14364 (2011).
- ⁸⁷W. Kohn, "Density functional and density matrix method scaling linearly with the number of atoms," *Phys. Rev. Lett.* **76**, 3168 (1996).
- ⁸⁸E. Prodan and W. Kohn, "Nearsightedness of electronic matter," *Proc. Natl. Acad. Sci. U. S. A.* **102**, 11635–11638 (2005).
- ⁸⁹J. P. Perdew, A. Ruzsinszky, J. Tao, V. N. Staroverov, G. E. Scuseria, and G. I. Csonka, "Prescription for the design and selection of density functional approximations: More constraint satisfaction with fewer fits," *J. Chem. Phys.* **123**, 062201 (2005).
- ⁹⁰A. A. M. Al-Saffar, H. Tao, and M. A. Talab, "Review of deep convolution neural network in image classification," in *2017 International Conference on Radar, Antenna, Microwave, Electronics, and Telecommunications (ICRAMET)* (IEEE, 2017), pp. 26–31.
- ⁹¹Q. Wu and W. Yang, "A direct optimization method for calculating density functionals and exchange–correlation potentials from electron densities," *J. Chem. Phys.* **118**, 2498–2509 (2003).
- ⁹²W. Yang and Q. Wu, "Direct method for optimized effective potentials in density-functional theory," *Phys. Rev. Lett.* **89**, 143002 (2002).
- ⁹³R. P. Feynman, "Forces in molecules," *Phys. Rev.* **56**, 340 (1939).
- ⁹⁴P. Pulay, "Ab initio calculation of force constants and equilibrium geometries in polyatomic molecules: I," *Mol. Phys.* **17**, 197–204 (1969).
- ⁹⁵L. A. Curtiss, K. Raghavachari, G. W. Trucks, and J. A. Pople, "Gaussian-2 theory for molecular energies of first- and second-row compounds," *J. Chem. Phys.* **94**, 7221–7230 (1991).
- ⁹⁶Y. Chen, L. Zhang, H. Wang, and E. Weinan, "DeePKS-kit: A package for developing machine learning-based chemically accurate energy and density functional models," *Comput. Phys. Commun.* **282**, 108520 (2023).
- ⁹⁷M. F. Kasim, S. Lehtola, and S. M. Vinko, "Dqc: A python program package for differentiable quantum chemistry," *J. Chem. Phys.* **156**, 084801 (2022).
- ⁹⁸H.-B. Ren, L. Wang, and X. Dai, "Differentiable programming and density matrix based Hartree–Fock method," *Chin. Phys. B* **30**, 060701 (2021).
- ⁹⁹S. Dick and M. Fernandez-Serra, "Highly accurate and constrained density functional obtained with differentiable programming," *Phys. Rev. B* **104**, L161109 (2021).
- ¹⁰⁰G. L. Oliver and J. P. Perdew, "Spin-density gradient expansion for the kinetic energy," *Phys. Rev. A* **20**, 397 (1979).
- ¹⁰¹E. H. Lieb and S. Oxford, "Improved lower bound on the indirect Coulomb energy," *Int. J. Quantum Chem.* **19**, 427–439 (1981).
- ¹⁰²K. Pokharel, J. W. Furness, Y. Yao, V. Blum, T. J. P. Irons, A. M. Teale, and J. Sun, "Exact constraints and appropriate norms in machine-learned exchange-correlation functionals," *J. Chem. Phys.* **157**, 174106 (2022).
- ¹⁰³R. Nagai, R. Akashi, and O. Sugino, "Machine-learning-based exchange correlation functional with physical asymptotic constraints," *Phys. Rev. Res.* **4**, 013106 (2022).
- ¹⁰⁴K. Burke, F. G. Cruz, and K.-C. Lam, "Unambiguous exchange-correlation energy density," *J. Chem. Phys.* **109**, 8161–8167 (1998).
- ¹⁰⁵W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*, 3rd ed.: The art of scientific computing (Cambridge University Press, 2007).
- ¹⁰⁶J. Kirkpatrick, B. McMorrow, D. H. P. Turban, A. L. Gaunt, J. S. Spencer, A. G. D. G. Matthews, A. Obika, L. Thiry, M. Fortunato, and D. Pfau, "Pushing the frontiers of density functionals by solving the fractional electron problem," *Science* **374**, 1385–1389 (2021).
- ¹⁰⁷J. Jaramillo, G. E. Scuseria, and M. Ernzerhof, "Local hybrid functionals," *J. Chem. Phys.* **118**, 1068–1073 (2003).
- ¹⁰⁸S. Grimme, J. Antony, S. Ehrlich, and H. Krieg, "A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H–Pu," *J. Chem. Phys.* **132**, 154104 (2010).
- ¹⁰⁹N. Tassinato and S. Grimme, "Unveiling the non-covalent interactions of molecular homodimers by dispersion-corrected DFT calculations and collision-induced broadening of ro-vibrational transitions: Application to (CH₂F₂)₂ and (SO₂)₂," *Phys. Chem. Chem. Phys.* **17**, 5659–5669 (2015).
- ¹¹⁰C. K. I. Williams and C. E. Rasmussen, *Gaussian Processes for Machine Learning* (MIT Press, Cambridge, MA, 2006), Vol. 2.
- ¹¹¹J. Proppe, S. Gugler, and M. Reiher, "Gaussian process-based refinement of dispersion corrections," *J. Chem. Theory Comput.* **15**, 6046–6060 (2019).
- ¹¹²J. P. Perdew, K. Burke, and M. Ernzerhof, "Generalized gradient approximation made simple," *Phys. Rev. Lett.* **77**, 3865 (1996).
- ¹¹³S. Grimme, S. Ehrlich, and L. Goerigk, "Effect of the damping function in dispersion corrected density functional theory," *J. Comput. Chem.* **32**, 1456–1465 (2011).
- ¹¹⁴F. Weigend and R. Ahlrichs, "Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy," *Phys. Chem. Chem. Phys.* **7**, 3297–3305 (2005).
- ¹¹⁵J. Zheng, X. Xu, and D. G. Truhlar, "Minimally augmented Karlsruhe basis sets," *Theor. Chem. Acc.* **128**, 295–305 (2011).
- ¹¹⁶C. Riplinger, B. Sandhoefer, A. Hansen, and F. Neese, "Natural triple excitations in local coupled cluster calculations with pair natural orbitals," *J. Chem. Phys.* **139**, 134101 (2013).
- ¹¹⁷C. Riplinger and F. Neese, "An efficient and near linear scaling pair natural orbital based local coupled cluster method," *J. Chem. Phys.* **138**, 034106 (2013).
- ¹¹⁸G. A. Petersson and M. A. Al-Laham, "A complete basis set model chemistry. II. Open-shell systems and the total energies of the first-row atoms," *J. Chem. Phys.* **94**, 6081–6090 (1991).
- ¹¹⁹D. G. A. Smith, L. A. Burns, K. Patkowski, and C. D. Sherrill, "Revised damping parameters for the D3 dispersion correction to density functional theory," *J. Phys. Chem. Lett.* **7**, 2197–2203 (2016).
- ¹²⁰T. Weymuth, J. Proppe, and M. Reiher, "Statistical analysis of semiclassical dispersion corrections," *J. Chem. Theory Comput.* **14**, 2480–2494 (2018).
- ¹²¹Z. Li, N. Kovachki, K. Azizzadenesheli, B. Liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Fourier neural operator for parametric partial differential equations," [arXiv:2010.00895](https://arxiv.org/abs/2010.00895) (2020).
- ¹²²N. Kovachki, Z. Li, B. Liu, K. Azizzadenesheli, K. Bhattacharya, A. Stuart, and A. Anandkumar, "Neural operator: Learning maps between function spaces," [arXiv:2108.08481](https://arxiv.org/abs/2108.08481) (2021).
- ¹²³D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," Technical Report (California University San Diego La Jolla Institute for Cognitive Science, 1985).

- ¹²⁴J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, and A. Potapenko, “Highly accurate protein structure prediction with AlphaFold,” *Nature* **596**, 583–589 (2021).
- ¹²⁵GitHub-Repo, “oep-wy-xcnn,” <https://github.com/zhouyyc6782/oep-wy-xcnn> (2021).
- ¹²⁶A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimeshin, L. Antiga *et al.*, “Pytorch: An imperative style, high-performance deep learning library,” in *Advances in Neural Information Processing Systems* [Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2019], Vol. 32.
- ¹²⁷Q. Sun, T. C. Berkelbach, N. S. Blunt, G. H. Booth, S. Guo, Z. Li, J. Liu, J. D. McClain, E. R. Sayfutyarova, S. Sharma *et al.*, “PySCF: The python-based simulations of chemistry framework,” *Wiley Interdiscip. Rev.: Comput. Mol. Sci.* **8**, e1340 (2018).
- ¹²⁸Y. Shi and A. Wasserman, “Inverse Kohn-Sham density functional theory: Progress and challenges,” *J. Phys. Chem. Lett.* **12**, 5308–5318 (2021).
- ¹²⁹A. Szabo and N. S. Ostlund, *Modern Quantum Chemistry: Introduction to Advanced Electronic Structure Theory* (Dover, 1989).
- ¹³⁰D. R. Hartree, “The wave mechanics of an atom with a non-Coulomb central field. Part I. Theory and methods,” *Math. Proc. Cambridge Philos. Soc.* **24**, 89–110 (1928).
- ¹³¹S. Amari, “Backpropagation and stochastic gradient descent method,” *Neuro-computing* **5**, 185–196 (1993).
- ¹³²D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” [arXiv:1412.6980](https://arxiv.org/abs/1412.6980) (2014).