CIDER: An Expressive, Nonlocal Feature Set for Machine Learning Density Functionals with Exact Constraints

Kyle Bystrom* and Boris Kozinsky*

ABSTRACT: Machine learning (ML) has recently gained attention as a means to develop more accurate exchange-correlation (XC) functionals for density functional theory, but functionals developed thus far need to be improved on several metrics, including accuracy, numerical stability, and transferability across chemical space. In this work, we introduce a set of nonlocal features of the density called the CIDER formalism, which we use to train a Gaussian process model for the exchange energy that obeys the critical uniform scaling rule for exchange. The resulting CIDER exchange functional is significantly more accurate than any semilocal functional tested here, and it has good transferability across main-group molecules. This work therefore serves as an initial step toward more accurate exchange functionals, and it also introduces useful techniques for developing robust, physics-informed XC models via ML.

1. INTRODUCTION

Density functional theory (DFT)\(^1\)\(^2\) is an indispensable tool in computational chemistry and materials science due to its combination of efficiency and accuracy. While an exact theory, the exchange-correlation (XC) functional describing quantum mechanical effects must be approximated in practice, and this approximation is the key limiting factor in the accuracy of DFT. Hundreds of approximate XC functionals have been developed in an attempt to calculate accurate chemical data,\(^3\)\(^4\)\(^5\)\(^6\)\(^7\)\(^8\)\(^9\)\(^10\)\(^11\)\(^12\) but significant progress remains to be made in the development of highly accurate XC functionals. Several reviews cover the shortcomings of current XC functionals,\(^13\)\(^14\)\(^15\)\(^16\)\(^17\)\(^18\)\(^19\)\(^20\)\(^21\) such as self-interaction error, overdelocalization, and inconsistent description of static correlation.

The shortcomings of existing approximations have motivated the development of machine learning (ML) models for more accurate functionals.\(^8\) This concept was first introduced by Tozer et al.\(^10\) and more recently pioneered by Burke and coworkers.\(^11\)\(^12\) Some of these models use projections of the electron density or density matrix onto atom-centered basis sets as input to a ML model, resulting in atomic decomposition of the XC energy.\(^13\)\(^14\)\(^15\) These models are highly accurate across a small set of systems similar to those on which the model is trained, but they do not match the universality of most conventional XC functionals. For example, DeePKS\(^14\) is only applicable to closed-shell organic molecules. Another recent approach uses a real-space grid and convolves the density to create features in real space.\(^16\)\(^17\) This approach has good transferability across chemical space and requires less training data than other ML functionals (as shown by Nagai et al.\(^17\) who used a training set of only three molecules), but the accuracy achieved thus far is not consistently better than conventional semiempirical XC functionals. For example, the NN-NRA by Nagai et al. is more accurate than M06-L\(^18\) and M06\(^19\) for main-group ionization potentials but less accurate for barrier heights.\(^17\)

The latter real-space approach is effectively an extension of the semiempirical approach to functional design, in which a parametric functional is fit to experimental or theoretical training data. There is a trade-off between designing functionals that fit experimental data well and functionals that obey analytically known constraints on the exact functional.\(^6\)\(^20\) It has been shown that semiempirical functionals have improved accuracy for energetic data, but breaking exact constraints can make the functional less universal and transferable, resulting in poor description of density distributions.\(^21\) Exact constraints for ML functionals have not received much attention thus far. However, one study on the 1D kinetic energy functional showed that the uniform scaling constraint improved the learning curve of a kernel ridge regression-based functional,\(^22\) and several constraints were enforced in a recent ML meta-GGA by Dick and Fernández-Serra.\(^23\) One can also explicitly train a ML model to reproduce the correct density; it has been shown that training

Received: September 8, 2021
Published: March 2, 2022
on both energetic data and density distributions can yield ML functionals that accurately predict both properties.\textsuperscript{14,17,23−25}

Comparing the various approaches above, it seems that the atomic decomposition approach lacks the physical intuition and data efficiency to develop broadly transferable functionals for the entire periodic table, whereas existing real-space approaches lack the flexibility to describe systems to a high degree of accuracy while maintaining sensible physical behavior like accurate density distributions. Therefore, we seek to develop a framework for constructing descriptors of the density distribution that can be used to design accurate and transferable xc functionals with ML. Such features must describe variations in the density that correlate with the target xc energy, but they do not need to have an intuitive physical meaning because the ML algorithm can learn the relationship between the features and the xc energy. Ideally, the features should enable a ML model trained on limited data to generalize across a broad range of chemistries.

For this work, we focus on the Kohn–Sham (KS) exchange functional\textsuperscript{26}

\begin{equation}
E_x^{KS}[n] = \langle \Phi^{KS}[n] | \hat{V}_e^{KS}[n] | \Phi^{KS}[n] \rangle - U[n]
\end{equation}

where \(\Phi^{KS}[n]\) is the KS Slater determinant, \(\hat{V}_e^{KS}[n] = \sum_i r_i^{-1}\) is the Coulomb repulsion operator, and \(U[n] = \int d^3r_1 \cdots d^3r_N n(r_1) \cdots n(r_N)\) is the classical Coulomb interaction. The rest of the paper is structured as follows. Section 2 introduces the CIDER formalism, and section 3 describes the Gaussian process models used with the CIDER descriptors to train exchange functionals. Section 4 describes the methods for performing the DFT calculations and training the CIDER functional, and section 5 contains the results and a discussion of the new functional’s performance across a diverse set of chemistries. Finally, section 6 concludes with a summary of the findings.

2. CIDER FORMALISM

To develop a ML model for the exchange energy, an expressive set of nonlocal descriptors of the density must be used as input to the model. One could use a neural network (NN) to learn the features from the raw density distribution in real space, but training features this way is data intensive, with \(10^5−10^6\) training points used in recent works.\textsuperscript{9,10} In addition, these NNs rely on a specific grid structure over which convolutions are performed, which could impede their use in realistic production calculations. Alternatively, one could project the density or density matrix onto atomic basis sets, as is done in NeuralXC and DeePKS,\textsuperscript{13,14} but these two models do not incorporate any physical constraints into the features, making it infeasible to incorporate exact constraints into the model itself. The difficulty of incorporating physical constraints and intuition into such models could limit their transferability and universality. A compromise between these two approaches is to design features based on nonlocal convolutions of the density, as done by Lei and Medford\textsuperscript{26} and by Nagai et al.,\textsuperscript{17} and then use these features as input to a ML model. We seek to improve on this third approach by designing descriptors that are highly expressive and which also constrain the resulting ML model to known properties of the exact functional.

The most important constraint for the exchange energy is derived from the principle of uniform scaling. Consider a density distribution \(n(r)\) and a scaled density

\begin{equation}
n_\gamma(r) = \gamma^3 n(\gamma r)
\end{equation}
Several important exact constraints can be written using this scaled density. They include the uniform scaling rules for the noninteracting kinetic energy \( T[n] \) and exchange energy \( E_x[n] \):

\[
T[n] = \gamma^2 T[n] \quad (3)
\]

\[
E_x[n] = \gamma E_x[n] \quad (4)
\]

Equation 4 implies that the exchange energy can be written as

\[
E_x[n] = -\frac{3}{2} \int d^3r F_x[n](r)n^{1/3}(r) \quad (5)
\]

which is the form used by most semilocal exchange functionals. The constant in front of the integral is chosen so that \( E_x = 1 \) for the homogeneous electron gas (HEG). The functional for a spin-unpolarized density \( E_x[n] \) can be extended to the spin-polarized case using the spin-scaling rule:

\[
E_x[n_\uparrow, n_\downarrow] = \frac{1}{2}(E_x[2n_\uparrow] + E_x[2n_\downarrow]) \quad (6)
\]

Because eq 6 uniquely and simply defines the spin-polarized exchange energy from the spin-unpolarized exchange energy, the remainder of the discussion in this section refers to the spin-unpolarized case.

The exchange enhancement factor (XEF) \( F_x[n](r) \) in eq 5 is independent of \( \gamma \), a property which will be referred to as scale invariance. Therefore, it is reasonable to predict that a ML model for \( F_x[n] \) will learn more efficiently if the feature vector \( \mathbf{x} \) used as input to the model is scale invariant (\( F_x[n](r) = \mathbf{x}(n)(r) \)). The conventional descriptors of the gradient \( \nabla n \) and kinetic energy density \( \tau = \frac{1}{2} \sum_i (|\nabla \phi_i|^2 \) satisfy these rules

\[
s = \frac{2(3\pi)^{1/3}}{2(3\pi)^{1/3}} n^{4/3} \quad (7)
\]

\[
\alpha = \frac{\tau - \tau_W}{\tau_0} \quad (8)
\]

where \( \tau_0 = |\nabla n|^2/8n \) is the kinetic energy density of a single-orbital system and \( \tau_W = (3/10)(3\pi)^{2/3} n^{0/3} \) is the kinetic energy density of the HEG. The descriptor \( \alpha \) was first introduced in by Sun et al.\(^{14} \) While these descriptors are useful, they are semilocal, so they cannot fully encode the complex, nonlocal structure of the exchange functional. The challenge is therefore to construct a set of nonlocal descriptors that are scale invariant, which can be used for accurately training a ML model of the functional \( E_x[n] \).

Nonlocality can be introduced to the features by performing convolutions on the density with a short-range kernel, as done in the near-region approximation (NRA) by Nagai et al.\(^{17} \) However, these features are not scale invariant. To achieve scale invariance, we use an approach similar to that developed by Janesko and co-workers for “Rung 3.5” semiempirical functionals.\(^{45-47} \) Rung 3.5 functionals use the one-particle density matrix \( n_1(r, r') \)

\[
n_1(r, r') = \sum_{j} f_j \phi_j(r) \phi_j(r') \quad (9)
\]

where \( \phi_j(r) \) are the Kohn–Sham orbitals and \( f_j \) are the occupation numbers (2 for the occupied orbitals in a spin-unpolarized system and 0 for unoccupied orbitals). The density matrix \( n_1(r, r') \) is projected onto a semilocal model for the density matrix at each point \( r \) to construct the Rung 3.5 energy density

\[
\varepsilon_{\text{Rung3.5}}[n](r) = \int d^3r_n(r, r')n_1(r) n_1(r - r') \quad (10)
\]

Importantly, the model density matrix \( n_1(r, r') \) is position dependent in a way that gives \( \varepsilon_{\text{Rung3.5}}[n](r) \) convenient scaling properties. For example, if \( n_1(r, r') \) is the exchange hole of the HEG then \( \varepsilon_{\text{Rung3.5}}[n](r) = \gamma \varepsilon_{\text{Rung3.5}}[n](r) \).

Following this approach, we introduce a scale-invariant set of integral descriptors to describe the density distribution around a point

\[
G_{\text{nlm}}(r) = \int d^3r \varepsilon_{\text{nlm}}(r; r)n(r + r') \quad (11)
\]

\[
\varepsilon_{\text{nlm}}(r; r') = \frac{3}{2} \sqrt{4\pi r l} \left( \frac{8\pi}{3} \right)^{1/3} Y_{lm}(\mathbf{r}) (|\mathbf{r}| / a)^{2l+1} e^{-a|\mathbf{r}|^2} \quad (12)
\]

In the above equation, the functions \( Y_{lm}(\mathbf{r}) \) represent the real spherical harmonics and the exponent \( a \) is a function of \( \mathbf{r} \) and a semilocal functional of the density

\[
n(r) \propto e^{-a|\mathbf{r}|^2} \quad (14)
\]

The \( B_0 = 1 \) case is important because the exchange hole integrates to \(-1 \) (−2 if summed over spin for a spin-unpolarized system), so for \( B_0 = 1 \), \( g_{000} \) is approximately shaped like the HEG exchange hole and has the correct norm for the exchange hole for the HEG. This gives the feature a sensible shape and length scale, and it also gives a known HEG reference value for that feature, allowing the HEG constraint to be enforced.

If \( B_0 = C_0 \) and \( a \) is finite in atomic core regions but will vanish at the center of single bonds, where \( \tau = 0 \). If \( C_0 = a \), \( a \) will vanish at the atomic tails as the density goes to zero. In each case, the length scale of the Gaussian distribution becomes infinite, which is incompatible with the goal of a computationally efficient functional and is also poor inductive bias because the exchange hole is localized around the reference point. Therefore, \( B_0 \) must be greater than \( C_0 \) and \( C_0 \) must be positive.

In this work, we use the constant \( A \) to define the length scale, such that

\[
B_0 = A \quad (15)
\]

\[
C_0 = \frac{A}{3} \frac{6}{325\pi} \approx 0.18A \quad (16)
\]

This choice was found to yield a smooth length scale in real space. We also define several choices of \( A \) for the model, resulting in different widths of the Gaussian kernels:

- **WIDE**, \( A = 1 \)
- **MEDIUM-WIDE**, \( A = 2 \)
- **MEDIUM-TIGHT**, \( A = 4 \)
- **TIGHT**, \( A = 8 \)
These nonlocal features, along with the semilocal features $s$ and $\alpha$, constitute the compressed scale-invariant density representation (CIDER), so named because the features satisfy $G_{nlm}[n]r(r) = G_{nlm}[n](\gamma r)$ and provide an efficient representation of the density distribution around a point $r$ in real space.

The XC energy is invariant under rotation and translation of the system. The raw CIDER descriptors above are translationally invariant but not rotationally invariant. To create rotationally invariant descriptors, contractions using Clebsch–Gordon coefficients are performed. This is similar to the approaches used in the tensor field network,48 the moment tensor potential,49 the field network,48 the moment tensor potential,49 the

In its current iteration, the CIDER descriptors are used to define the feature vector $x$, where $G_{nlm}$ is a vector containing $G_{nlm}$ for $m \in \{-2l-1, -2l, ..., 2l+1\}$

\[ x_1 = s^2 \]
\[ x_2 = \alpha \]
\[ x_3 = G_{00} \]
\[ x_4 = |G_{01}|^2 \]
\[ x_5 = |G_{02}|^2 / \sqrt{S} \]
\[ x_6 = s \cdot G_{01} \]
\[ x_7 = C(s, G_{02}, s) \]
\[ x_8 = C(s, G_{02}, G_{01}) \]
\[ x_9 = G_{00}(2^{-4/3}A) \]
\[ x_{10} = G_{00}(2^{4/3}A) \]  
(17)

$G_{00}(\lambda A)$ indicates that the length-scale parameter $A$ was changed to $\lambda A$ for this integral. $C(a, b, c)$ contracts the two $l = 1$ terms $a, c$ and the $l = 2$ term $b$ to an $l = 0$ term using Clebsch–Gordon coefficients (see Appendix B for details). For length-scale parameter $A$, eq 17, contains all possible rotationally invariant contractions of features with $l \leq 2$ with the exception of $x_{11} = C(G_{11}, G_{00}, G_{00})$. We found that this set of features provides a good balance between computational efficiency and model expressiveness. In principle, one could expand the feature set by including contractions of features with $l > 2$ and using multiple different length-scale parameters, but to keep the computational cost of feature evaluation tractable, this is not done here. One could also attempt to learn on smaller feature sets, including a semilocal model containing only $x_1$ and $x_2$. However, as discussed in the Supporting Information (section S2.2), this results in insufficiently accurate models. In section S2.2, we also show that adding the feature $x_{11}$ does not improve the accuracy, and excluding the additional $l = 0$ features $x_3$ and $x_{10}$ is detrimental to accuracy. We therefore conclude that eq 17 is a reasonable choice of feature vector.

The above descriptors will be referred to as Version A. We also tried introducing $G_{nlm}$ descriptors for $n \neq 0$, yielding Version C (Compact) descriptors, which are the same as Version A except for the last two

\[ x_9 = G_{10} \]
\[ x_{10} = G_{00}(2A) \]  
(18)

\[ K_{ij}^{(C)}(\mathbf{x}_i, \mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j) - k(\mathbf{x}_i, \mathbf{0}) - k(\mathbf{0}, \mathbf{x}_j) + k(\mathbf{0}, \mathbf{0}) \]  
(19)

While $x_{11}$ is a "tight" descriptor like in Version A, $x_9$ has the same exponent as the other nonlocal features but is multiplied by $\alpha^2$. This version allows us to examine the effects of eliminating the widest feature (thus "Compact"), which is useful because shorter range features might be more amenable to computationally efficient evaluation in optimized implementations of the model.

Normalizing features is generally helpful in machine learning applications. Using the descriptors developed by Becke,50 Becke and Edgecombe,51 and Mardirossian and Head-Gordon52 as guidance, we apply the following transformations to the above descriptors

\[ x_1 \rightarrow \frac{y_{x}x_{1}}{1 + y_{x}x_{1}} \]
\[ x_2 \rightarrow \frac{2}{1 + x_{2}^2} - 1 \]
\[ x_3 \rightarrow \frac{\gamma_{00}x_{3}}{1 + \gamma_{00}x_{3}} - c_{3} \]
\[ x_4 \rightarrow \frac{y_{x}x_{4}}{1 + y_{x}x_{4}} \]
\[ x_5 \rightarrow \frac{\gamma_{x}x_{5}}{1 + \gamma_{x}x_{5}} \]
\[ x_6 \rightarrow \frac{\gamma_{x}x_{6}}{\sqrt{1 + y_{x}x_{1}}} \frac{\gamma_{x}x_{1}}{\sqrt{1 + y_{x}x_{4}}} \]
\[ x_7 \rightarrow \frac{\gamma_{x}y_{x}}{1 + y_{x}x_{1}} \frac{\gamma_{x}y_{x}}{1 + y_{x}x_{4}} \]
\[ x_8 \rightarrow \frac{\gamma_{00}y_{x}}{1 + \gamma_{00}x_{9}} - c_{9} \]
\[ x_{10} \rightarrow \frac{\gamma_{00}x_{10}}{1 + \gamma_{00}x_{10}} - c_{10} \]  
(19)

The resulting transformed descriptors all fall in a finite range, making them more convenient for ML models. The hyperparameters in the equations above were selected heuristically as described in the Supporting Information (section S1), but they could also be optimized, if desired, by treating them as hyperparameters of Gaussian process regression models, as explained in section 3. The constants $c_9$ and $c_{10}$ guarantee that the zero feature vector $x = 0$ corresponds to the HEG. The HEG limit can therefore be enforced by setting $F_x = 1$ for $x = 0$ in eq S.

### 3. GAUSSIAN PROCESS EXCHANGE MODELS

To train ML models of the functionals, we employ Gaussian processes (GPs), which are commonly used for nonparametric regression in Bayesian statistical learning models.53 For a training set size $N$, matrix of inputs $X$ (the set of feature vectors $x^{(i)}$ for the training points $i \in \{1, ..., N\}$), vector of outputs $y$, and kernel function $k(x, x')$, the standard GP relations for the predictive mean $f(x)$ and variance $\sigma^2$ are

\[ f(x) = k(x, \mathbf{y}) (\mathbf{K} + \sigma^2_n \mathbf{I})^{-1} \mathbf{y} \]
\[ \sigma^2(x) = k(x, x') - k(x, \mathbf{0}) - k(\mathbf{0}, x') + k(\mathbf{0}, \mathbf{0}) \]  
(20)

\[ f(x) = k(x, \mathbf{y}) (\mathbf{K} + \sigma^2_n \mathbf{I})^{-1} \mathbf{y} \]
\[ \sigma^2(x) = k(x, x') - k(x, \mathbf{0}) - k(\mathbf{0}, x') + k(\mathbf{0}, \mathbf{0}) \]  
(21)
In the above equation, \( x_* \) is the test point, \( k_* \) is a vector containing \( k(x_i^{(0)}, x_*) \) for each \( x_i^{(0)} \) in \( X \), \( I \) is the identity matrix, and \( K_{ij} = k(x_i^{(0)}, x_j^{(0)}) \) for each \( x_i^{(0)}, x_j^{(0)} \) in \( X \). \( k(x, x') \) can be any function satisfying the rules for an inner product, and it may contain a set of hyperparameters \( \theta \) that may need to be optimized. These hyperparameters can be chosen by maximizing the marginal likelihood

\[
\ln p(y|X, \theta) = - \frac{1}{2} y^T (K + \sigma_{\text{noise}}^2 I)^{-1} y - \frac{1}{2} \ln |K + \sigma_{\text{noise}}^2 I| - \frac{N}{2} \ln 2\pi
\]

(22)

The noise parameter \( \sigma_{\text{noise}} \) can also be optimized in this manner. One popular covariance kernel is the squared exponential kernel

\[
k(x, x') = \exp \left( -\frac{1}{2} ((x - x') \otimes a^2) \right)
\]

(23)

where \( \otimes \) represents the element-wise product and \( a \) is a hyperparameter vector containing the inverse standard deviation in each direction in feature space. This kernel is used for some of the preliminary models discussed in the Supporting Information (section S1) and as the base kernel for the additive kernel discussed below. The squared exponential kernel yields highly expressive, nonlinear models because it is equivalent to linear regression on an infinite set of nonlinear basis functions.\(^{53,55}\) It is also smooth and infinitely differentiable, which is important for ML functional models because they must be differentiable to obtain the exchange potential.

One drawback of the GP is that evaluating eq 20 for a single test point scales linearly with the training set size \( N \), so evaluating GPs with large training sets is computationally expensive. To enable efficient evaluation of the model, some developers of GP-based molecular dynamics force fields map the GP to a cubic spline of the features, so evaluation of the model has a low cost that is independent of training set size.\(^ {54-55}\) However, this approach is only feasible for \( n \leq 4 \) features because the amount of memory required for the spline coefficients scales as \( G^n \), with \( G \) being the grid size in each dimension. With the 10 features in the CIDER model, eq 23 cannot be mapped to a cubic spline.

This problem can be solved by the additive kernel developed by Duvenaud et al.\(^ {58}\)

\[
k_{\text{add}}(x, x') = \sigma_{\text{noise}}^2 \sum_{1 \leq i_1 < i_2 < \cdots < i_D \leq n} \prod_{d=1}^D k_i(x_{i_d}, x'_{i_d})
\]

(24)

where \( D \) is the dimensionality of the feature vector, \( n \) is the order of the kernel, and \( k_i(x_{i_d}, x'_{i_d}) \) is the kernel for feature \( i \). For this kernel, the predictive mean can be linearly decomposed into terms with a maximum of \( n \) features per term. Each of these terms can be mapped to a cubic spline in a memory-efficient manner for \( n \leq 4 \). See Appendix C for a more detailed explanation.

### 3.1. Model Details

The additive Gaussian process approach allows interactions of any order in the descriptors to be accounted for by changing \( n \) in eq 24. Using the feature vector \( x \) of eq 17, the kernel for the exchange matrix is constructed by multiplying the \( x_i \) kernel by an \( n = 2 \) additive kernel for the other 9 descriptors

\[
k_{\text{CIDER}}(x, x') = \sigma_{\text{noise}}^2 k_i(x_i, x_i) + \sigma_{\text{noise}}^2 \sum_{i=2}^{10} k_i(x_i, x_i') + \sigma_{\text{noise}}^2 \sum_{i=2}^{10} \sum_{j=i+1}^{10} k_i(x_i, x_j)k_i(x_j, x_j')
\]

(25)

where \( k_i(x, x') = \exp \left( -\frac{1}{2} d^2(x - x')^2 \right) \) is a squared exponential kernel. The exponent hyperparameters \( a_i \) and weights \( \sigma_n \) are fit using iterative maximum likelihood optimization. In practice, \( \sigma_1 \) and \( \sigma_2 \) vanish during hyperparameter optimization. The resulting predictive mean is mapped onto a sum of cubic splines, as described in previous work on molecular dynamics potentials.\(^ {54,55}\) Because the maximum number of features per term in eq 25 is 3, the cubic splines are at most three dimensional.

We train our predictive function \( F^\text{pred}(x[n](r)) \) to the XEF, \( F^\text{obs}[n](r) \), and evaluate \( F[n](r) \) via eq 5

\[
F^\text{obs}[n](r) = \frac{e^\text{ex}[n](r)}{e^\text{LDA}[n](r)}
\]

(26)

\[
e^\text{ex}[n](r) = -\frac{1}{4} \int d^3 r_1 \ln \left[ |r - r_1|^2 \right]
\]

(27)

\[
e^\text{LDA}[n](r) = -\frac{3}{4} \left( \frac{3}{\pi} \right)^{1/3} n^{4/3}
\]

(28)

In eq 27, \( e^\text{ex}[n](r) \) is the exact exchange energy density and \( n(r, r') \) is the density matrix (eq 9). Equation 27 is not a unique definition, as any \( e^\text{ex}[n](r) \) for which

\[
E[n] = \int d^3 r e^\text{ex}[n](r)
\]

(29)

is equally valid, but eq 27 is the easiest to compute and was found to be the easiest to which to fit the model as well. The Supporting Information (section S4) includes a discussion of alternative exchange energy densities. One of the challenges with the definition in eq 27 is that the XEF increases exponentially at the atomic tails. To make the model easier to train, the Chachiyo GGA exchange functional,\(^ {59}\) which reproduces \( F^\text{ex} \) at atomic tails, was used as a baseline functional, and the difference \( F[n](r) - F^\text{Chachiyo}[n](r) \) was learned.

The standard GP model uses a single noise parameter \( \sigma_{\text{noise}} \) which is constant for all observations. However, we expect observations of \( F^\text{ex} \) to have a larger uncertainty for smaller densities because the exchange energy density is divided by \( (n(r))^{4/3} \). Therefore, we use a heteroskedastic noise model in which each training point has a different noise parameter. This noise parameter is a function of the density \( n(r) \) and takes the form

\[
\sigma_{\text{noise}}^2(n) = \nu_1 + \frac{\nu_2}{1 + t_1 n} + \frac{\nu_3}{1 + t_2 n}
\]

(30)

where \( t_1 = 50 \), \( t_2 = 10^6 \), and \( \nu_1 \), \( \nu_2 \), and \( \nu_3 \) are treated as hyperparameters and optimized by marginal likelihood maximization.

For numerical stability, the ML part of the functional is cut off at low densities. This is done using the function

\[
F^\text{x} = F^\text{Chachiyo} + F^\text{ML}
\]

(31)
where

\[ f_{\text{cut}} = \begin{cases} 
0 & n < n_{\text{r}, \text{min}} \\
\frac{1}{2} \left( 1 - \cos \left( \pi f_{\text{cut}} \right) \right) & n_{\text{r}, \text{min}} \leq n \leq n_{\text{r}, \text{max}} \\
1 & n > n_{\text{r}, \text{max}} 
\end{cases} \quad (32) \]

\[ x_{\text{cut}} = \frac{\ln(n_{\text{r}}/n_{\text{r}, \text{min}})}{\ln(n_{\text{r}, \text{max}}/n_{\text{r}, \text{min}})} \quad (33) \]

\[ n_{\text{r}} = \max(n, n_{\text{r}, \text{max}}) \quad (34) \]

where \( n_{\text{r}, \text{max}} = 10^{-3} \) and \( n_{\text{r}, \text{min}} = 10^{-6} \) Bohr\(^{-3}\).

Also, for numerical stability, we define a value \( a_{\text{cut}} \) below which the exponent \( a(r) \) in eq 13 is exponentially damped so that it cannot go to zero

\[ a \rightarrow \begin{cases} 
a & a \geq a_{\text{cut}} \\
a_{\text{cut}} \exp(a/a_{\text{cut}} - 1) & a < a_{\text{cut}} \end{cases} \quad (35) \]

\( a_{\text{cut}} \) is set to \( A/16 \) for \( A \) in eq 15. This damping of the exponent violates the uniform scaling rule but only at low densities for very diffuse orbitals.

4. METHODS

4.1. Computational Details. The PySCF\(^{60}\) code was used for all calculations. The CIDER implementation is available in the CiderPress repository.\(^61\) To compute the CIDER descriptors, we first projected the density matrix onto the def2-Coulomb fitting auxiliary basis.\(^62\) Then, the overlaps of the descriptor functions \( g_{\text{aux}}(r; r') \) with the auxiliary basis were computed using the libint\(^63\) library as interfaced through PySCF. These overlaps were contracted with the density to give the descriptors \( G_{\text{aux}}(r) \), which were then used to compute the XEF. Then, the functional derivatives were computed and used to evaluate the exchange matrix elements. The functional was treated self-consistently except for the \( f_{\text{cut}} \) function in eq 32, which was not differentiated with respect to the density. The formulas for the functional derivatives and matrix elements are written in Appendix D. The fireworks\(^64\) package was used to automate calculation workflows.

4.2. Training and Testing the Gaussian Process. The data set developed by Jensen et al.\(^65\) was used to train, validate, and test the exchange model. This data set is an expansion of the G2/97\(^{2197}\) test set\(^66\) that includes elements that are under-represented in the G2/97 set as well as six nonbonded systems.\(^67\) The molecules were categorized by the elements they contained (Al, Be, B, Li, Mg, Na, P, Si, S, F, Cl), with the earlier elements in the list taking precedence (e.g., SF\(_6\) was categorized under S). The last set contained mostly organic molecules with none of these elements. Each subgroup was duplicated P\(_2\) molecule. Both duplicates were placed in the test set, so one was simply removed. The final partitions contained 79 training set molecules (55 closed shell, 24 open shell), 42 validation set molecules (29 closed shell, 13 open shell), and 90 test set molecules (64 closed shell, 26 open shell). In addition to these molecules, the training set was augmented with isolated atoms H−Kr as well as the excited spin states of Sc, Ti, V, and Cr with \( 2S = 3, 4, 5, \) and 4, respectively. The inclusion of isolated atoms helped better describe the shapes of the tails of atoms, and it also introduced some transition-metal atoms to the training set. There were no transition-metal-containing systems in the Jensen data set.

The density matrices and energies for each system were evaluated self-consistently using PBE\(^{68}\) in the def2-QZVPPD basis set. From the PBE density matrix, the conventional exchange energy density (eq 27) was calculated in real space for each molecule on the level-3 grids implemented in PySCF. The same level-3 grids were used for all SCF calculations. The \( S = 0 \) systems were performed in the Restricted Kohn−Sham (RKS) formalism, and the rest were performed in the unrestricted Kohn−Sham (UKS) formalism. From the real-space grids, 14,695 points were sampled from which to train the exchange enhancement factor model

- 481 points from closed-shell atoms,
- 4471 points from open-shell atoms,
- 6903 points from closed-shell molecules in the Jensen train partition,
- 2840 points from open-shell molecules in the Jensen train partition.

The training set size of 14,695 was chosen to maximize the training set size while keeping the computational cost tractable. For comparison, we benchmarked the train and validation performance of the CIDER functional for different training set sizes in Supporting Information section S2.1 and found only a weak dependence of the accuracy on training set size. The points were selected randomly from the set of points for which the electron density was greater than \( 10^{-6} \) Bohr\(^{-3}\). Finally, the HEG limit was appended as an additional training point with the density set to \( 10^8 \) to minimize the uncertainty in eq 30. In summary, the CIDER model was trained to the exact exchange energy densities of density matrices obtained from self-consistent PBE calculations.

Gaussian process models with the kernel specified in eq 25 were trained to the training set described above. The parameters for the kernels were selected as described in detail in the Supporting Information (section S1). Several combinations of exact constraints, descriptor types, and descriptor length scales were tested, but we focused on three for the main body of this work.

- CIDER-X-CHW: Same as CIDER-X-AHW but with Version C descriptors.
- CIDER-X-CHMT: Same as CIDER-X-CHW but with MEDIUM-TIGHT descriptors (\( A = 4 \) in eq 15).

All models were implemented in the scikit-learn package,\(^69\) the additive RBF was implemented as a custom Kernel object. The additive kernel models were mapped to cubic splines using the interpolation.py package.\(^70\) The best-performing model on the validation set (CIDER-X-AHW) was selected as the final model.

To evaluate the accuracy and transferability of CIDER-X-AHW on static densities, the exchange energies and atomization energies were evaluated on the test set PBE density matrices using CIDER-X-AHW. To evaluate the self-consistent performance of the functional, the atomization energies of the Jensen test set were computed self-consistently using two functionals: CIDER-X-AHW alone (i.e., replacing exact exchange with CIDER in an
4. RESULTS AND DISCUSSION

5.1. Static Density Benchmark. As shown in Figure 1, three different versions of the CIDER functional all outperform existing semilocal functionals for predicting the exact exchange energy of PBE density matrices. We chose to compare with existing semilocal functionals for predicting the exact exchange energy because of their popularity, with the accuracy of PBE density matrices. The Supporting Information (Table S1) contains the validation set errors for several other semilocal functionals, which perform similarly to the functionals discussed here.

The best-performing CIDER functional in Figure 1 is CIDER-X-AHW, with an RMSE on the test set of 12.0 kcal/mol for exchange energy (EX) and 12.5 kcal/mol for atomization exchange energy (AEX), which is defined as the difference between EX and the value of EX for the molecule’s constituent atoms. These errors are only 18% (EX) and 27% (AEX) of the error (RMSE) of the most accurate semilocal functional, the Chachiyo GGA.

The other notable finding in Table 1 is that longer length-scale descriptors (CIDER-X-AHW) do not improve the description of EX over shorter length-scale descriptors (CIDER-X-CHMT), but they do give an improvement for AEX. In particular, the accuracy of AEX degrades as the length scale gets shorter, while the accuracy of EX stays about the same. This suggests that a descriptor with a short length scale cannot describe the localized atomic density and the more delocalized molecular density simultaneously. On the other hand, descriptors with longer length scales can accurately describe both single-center and multicenter exchange holes. Because of its accuracy on the
validation set for both EX and AEX, CIDER-X-AHW was used for the self-consistent field calculations below. The adjustable parameters for eqs 19 and 30 for CIDER-X-AHW are $\gamma_1 = 1/2$, $\gamma_2 = 0.0025$, $v_1 = 0.015$, $v_2 = 10^{-6}$, $v_3 = 0.000503$, and $v_4 = 0.391$.

5.2. Self-Consistent Field Calculations with B3LYP-CIDER. To be useful, the CIDER exchange functional must not only outperform semi-local exchange functionals but also accurately match the results of calculations performed using HF or hybrid functionals. To test this, SCF calculations were performed using CIDER-X-AHW and B3LYP-CIDER. Table 1 gives the mean absolute error (MAE) and RMSE of these functionals compared to HF and B3LYP, respectively.

To be useful, the CIDER exchange functional must not only outperform semi-local exchange functionals but also accurately match the results of calculations performed using HF or hybrid functionals. To test this, SCF calculations were performed using CIDER-X-AHW and B3LYP-CIDER. Table 1 gives the mean absolute error (MAE) and RMSE of these functionals compared to HF and B3LYP, respectively.

Before examining these results, it is worth noting that KS exact exchange (against which CIDER is trained) and HF exact exchange (against which CIDER is tested for SCF calculations) are different quantities yielding different effective potentials.26 To illustrate why this is, consider that in mean-field theory the Hamiltonian matrix elements ($\mu \| \nu$) must be evaluated for some basis set ($\chi_{\mu}(r)$) (assumed to be real for simplicity). The KS and HF exchange potential matrix elements are, respectively

$$\langle \mu | v_{x}^{KS} | \nu \rangle = \int d^{3}r \chi_{\mu}(r) \chi_{\nu}(r) \frac{\partial [n]}{\partial n(r)}$$

(36)

$$\langle \mu | v_{x}^{HF} | \nu \rangle = -\frac{1}{2} \int d^{3}r d^{3}r' \chi_{\mu}(r) \chi_{\nu}(r') \frac{n_{i}(r, r')}{|r - r'|}$$

(37)

These two forms of the exchange matrix elements are distinct. Therefore, the HF and KS exchange energies, potentials, and densities are different, though this difference tends to be small (about 0.03–0.04% of the total exchange energy for isolated atoms).26 Because exact computation of $\frac{dE_{\|d}}{dn(x)}$ is complicated and computationally expensive,85 most modern hybrid DFT calculations (including those performed in this work) use the Generalized Kohn–Sham (GKS) scheme,86,87 in which the HF matrix elements are used for the exact exchange potential instead of the pure KS matrix elements. Because the difference between HF and KS exchange is small and because evaluating $\frac{dE_{\|d}}{dn(x)}$ is difficult, it is reasonable to compare CIDER exchange to HF exchange. A more detailed explanation of exchange functionals in the GKS scheme is provided in Supporting Information section S5 using the Levy–Lieb constrained search formalism.88–90

As shown in Table 1, CIDER-X-AHW has an RMSE of 12 kcal/mol, which is fully explained by the RMSE of the CIDER-X-AHW predictions for static densities of 14 kcal/mol (Figure 1). It is notable that the SCF atomization energy errors slightly improve on the atomization exchange energy errors for static densities, even though the functional was only trained on static densities. This suggests that the CIDER exchange potential is sensible.

The 14 kcal/mol error of the CIDER-X-AHW functional with respect to exact exchange is small compared to that of the semi-local exchange functionals investigated here (Figure 1) but large compared to the desired chemical accuracy of 1 kcal/mol. However, hybrid functionals often use a small fraction of exact exchange, e.g., 20% for B3LYP. Because of this, B3LYP-CIDER reproduces B3LYP with an RMSE of only 2.3 kcal/mol on the test set (Table 1). While greater than 1 kcal/mol, this error is promisingly small considering that many of the atomization energies in the Jensen database involve breaking several chemical bonds at once.

To test the accuracy of the CIDER functional for both properties and systems significantly different than those contained in the training set, B3LYP and B3LYP-CIDER were used to compute the barriers heights of the BH76 database, the ionization potentials of the IP23 database, and the transition-metal bond energies of the TMBE33 database. These databases are subsets of the Minnesota 2015 Database.71 The MAEs for the transition-metal bond energies in Figure 2 are divided by the average number of bonds broken per data point, which matches the presentation in the original work.71

As shown in Figure 2, B3LYP-CIDER exacerbates the systematic underestimation of barrier heights of B3LYP, leading to an increase in MAE from 4.1 to 6.8 kcal/mol compared to the reference values. The MAE between B3LYP-CIDER and B3LYP is 2.8 kcal/mol. The deviation from B3LYP could potentially be improved by including the density distributions of transition states in the CIDER training set, as currently all training set systems are isolated atoms or molecules at equilibrium geometry.

For ionization potentials, B3LYP-CIDER performs well; the MAE compared to B3LYP is 1.7 kcal/mol, and the error compared to reference values is worse by only 0.5 kcal/mol. Most of the error compared to B3LYP arises from the transition-metal systems. For the main-group IP13-MG data set, B3LYP-CIDER reproduces B3LYP to chemical accuracy (0.8 kcal/mol), while the functionals deviate by 3.0 kcal/mol for the transition-metal IP10-TM subset. The chemically accurate reproduction of main-group IPs is notable because B3LYP-CIDER was not trained on any ionic systems. It might be that the uniform scaling rule allows the functional to relate ionic densities to atomic ones, e.g., relating C+ to B because they have the same shell structure. For transition-metal bond energies, B3LYP-CIDER has an MAE of 8.7 kcal/mol relative to B3LYP. However, B3LYP-CIDER is more accurate than B3LYP compared to the reference values; the error of the CIDER functional seems to cancel some systematic error of B3LYP for metal bond energies. It is known to be challenging to describe transition-metal bonds with hybrid...
functionals because the semilocal functional’s cancellation of error is lost. While ideally the CIDER functional would describe the exchange energy accurately, it is reassuring to see that it behaves like a semilocal functional when its accuracy breaks down, leading to sensible and explainable behavior.

Of note, the IP10-TM and TMBE33 data sets include ions, bonded transition-metal systems, and fifth-period elements treated with effective core potentials, none of which are present in the training set for CIDER. While CIDER is clearly a less faithful reproduction of HF for these systems than for systems similar to the training set, the resulting errors compared to reference values are similar. In addition, other than the isolated Fe atom and Pd’ ion, which had to be treated with level shifting and a high damping factor, calculations using the CIDER exchange functional converged successfully, as shown in Table 2. This suggests that the structure of the CIDER functional encourages transferability and stability.

### Table 2. Threshold to Which B3LYP-CIDER SCF Calculations Converged (in Ha) for the 451 Systems Involved in This Study

<table>
<thead>
<tr>
<th>Convergence threshold</th>
<th>No. of systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10^{-9}$</td>
<td>425</td>
</tr>
<tr>
<td>$10^{-8}$</td>
<td>18</td>
</tr>
<tr>
<td>$10^{-7}$</td>
<td>3</td>
</tr>
<tr>
<td>$10^{-6}$</td>
<td>2</td>
</tr>
<tr>
<td>Unconverged</td>
<td>451</td>
</tr>
</tbody>
</table>

“The two “unconverged” calculations were the Pd’ ion and Fe atom, which would only converge to $10^{-7}$ Ha when level shifting was applied.

Lastly, it is worth noting that the CIDER-X-AHW functional was not trained to any total energies, only exchange energy densities. It is possible that restraining the coefficients of hybrid functionals specifically for use with CIDER could significantly improve their accuracy, especially when the functional includes a higher fraction of exact exchange or already has a systematic error for some properties or systems (like B3LYP for barrier heights). An additional consequence of training to the exchange energy density is that it is nontrivial to extend the methodology presented here to the correlation functional. This is because the correlation energy density would be much harder to compute and more ambiguously defined than the exchange energy density. However, this is a limitation of the current Gaussian process model and training procedure, not of the CIDER features themselves. To train a correlation functional, one could either develop an approach to train to total correlation energies with a Gaussian process or use a model for which it is easier to train to total energies, such a neural network.

### 5.3. Comment on Computational Cost

The cost of evaluating the ML model is insignificant because it is mapped to a cubic spline, and the computational bottleneck is the evaluation of the features. Because this initial CIDER model has a large feature set, uses a relatively dense integration grid, and lacks matrix element screening for scalability, feature evaluation is slow for practical applications. However, all of the nonlocal features used in the model are orbital independent and have a finite length scale, suggesting that linear scaling and efficient implementations are possible. The challenge of optimizing CIDER functionals to have near-semilocal DFT cost will be the subject of future work. As a first step toward improving the computational efficiency, the Supporting Information (section S3) introduces a prospective algorithm for evaluating CIDER functionals in a linear-scaling fashion. This algorithm will be implemented in a future work.

### 6. CONCLUSION

In this work, we presented the CIDER formalism, which is based on a set of nonlocal features to describe the density distribution in a scale-invariant manner. This feature set was used to train a Gaussian process regression model to accurately describe the Kohn–Sham exchange functional $E_{\text{x}}[\rho]$ to a level of precision previously only attained by exact evaluation of the functional. The CIDER functional can replace a small fraction of HF exchange in hybrid functionals to accurately reproduce atomization energies, and it has excellent numerical stability, which has previously been a challenge for ML functionals.

As it stands, CIDER demonstrates that smooth, numerically stable exchange functionals can be learned that satisfy known exact constraints, accurately reproduce the target energy, and can be applied across a broad range of the periodic table. This demonstration is a first step toward a functional that could help bridge the gap between efficient semilocal functionals and accurate hybrid functionals as well as provide a groundwork for developing XC functionals with posthybrid DFT accuracy.

### APPENDIX A: RELATIONSHIP BETWEEN CIDER LENGTH SCALE AND SLATER ORBITAL DENSITY

Consider eq 13 in the case that $B_0 = C_0 = \frac{6}{8\pi} (6\pi^2)^{2/3}$. Then, noting that $\tau_0 = \frac{3}{10} (3\pi^2)^{2/3} n^{5/3}$, eq 13 becomes

$$a(n)(r) = \left( \frac{n}{2} \right)^{2/3} C_0 \frac{\tau}{\tau_0} \tau$$  \hspace{1cm} (38)

$$= 4\pi \frac{n}{n}$$  \hspace{1cm} (39)

Now suppose that the density distribution is a spin-unpolarized, two-electron system, in which case $\tau = \frac{\sqrt{n}}{8\pi}$. Then

$$a(n)(r) = \frac{1}{2} \left( \frac{\nabla n}{n} \right)^2$$  \hspace{1cm} (40)

If the electron pair occupies a Slater-type orbital then

$$n(r) = \frac{\sigma^3}{4\pi} e^{-\sigma r}$$  \hspace{1cm} (41)

for some $\sigma$. This orbital has $|\nabla n|/n = \sigma$, so

$$a(n)(r) = \sigma^2 \frac{2}{2}$$  \hspace{1cm} (42)

Then, it holds from eqs 41 and 42 that $n(r) \propto e^{-(\sqrt{2} \sigma)}$, as asserted in eq 14.

### APPENDIX B: CONTRACTING TWO $L = 1$ FEATURES AND ONE $L = 2$ FEATURE INTO AN $L = 0$ FEATURE

Take tensor features $a$, $b$, and $c$, with $l = 1$, 2, and 1, respectively, defined with real spherical harmonics $x_1, y_1, z_1$ for $l = 1$ and $x_2, y_2, z_2$ for $l = 2$. We can convert these features to and from the space of complex spherical harmonics using the following rules, with $Y_{lm}$ for the real spherical harmonics and $\overline{Y}_{lm}$ the complex ones.
evaluation easy and brings the computational cost of evaluation down to $O(1)$ per test point.

## APPENDIX D: FUNCTIONAL DERIVATIVES FOR CIDER NONLOCAL FEATURES

The Gaussian process is a function of a set of features

$$e_c[n](r) = -\frac{3}{4}\left(\frac{3}{\pi}\right)^{1/3} \left(n(r)\right)^{4/3} F_3(x[n](r))$$

with the total exchange energy being

$$E_x[n] = \int d^3 r e_c[n](r)$$

For the remainder of this section, we denote terms like $x[n]$ as $x$ for brevity. Calculating the exchange potential requires functional derivatives with respect to $n(r), \partial_\alpha n(r)$ for $\alpha = x, y, z$, and $r(r)$. Existing routines in PySCF and other DFT codes can compute the Generalized Kohn–Sham potential from the functional derivatives with respect to these quantities. These functional derivatives can be written as

$$\frac{\delta E_x}{\delta n(r)} = -\frac{3}{4}\left(\frac{3}{\pi}\right)^{1/3} \left(n(r)\right)^{4/3} \sum_i \frac{\partial E}{\partial x_i} \left|_{x(r)} \right| \frac{\delta x_i(r)}{\delta n(r)}$$

$$\frac{\delta E_x}{\delta \partial_\alpha n(r)} = -\frac{3}{4}\left(\frac{3}{\pi}\right)^{1/3} \int d^3 r n(r)^{4/3} \sum_i \frac{\partial E}{\partial x_i} \left|_{x(r)} \right| \frac{\delta x_i(r)}{\delta \partial_\alpha n(r)}$$

The terms $\frac{\partial E}{\partial x_i}$ are provided by the Gaussian process or cubic spline. If the index $i$ corresponds to a semilocal descriptor then

$$\frac{\delta x_i(r)}{\delta n(r)} = \left. \frac{\delta x_i(r)}{\delta n(r)} \right|_{n(r)}$$

and the integral over $r'$ reduces to evaluating the derivatives at $r$.

For the CIDER model, the $G_{\text{slim}}$ descriptors (eq 11) only have nonlocal dependence on the density $n(r)$; the dependence on $r(r)$ is local, and $G_{\text{slim}}$ does not depend on $\partial_\alpha n(r)$. Therefore, eq 59 applies with $r(r)$ in place of $n(r)$, even if $x_i$ is nonlocal. The kinetic term $\frac{\partial G_{\text{slim}}}{\partial a} = \frac{\partial G_{\text{slim}}}{\partial r} \left|_{r(r)} \right| \frac{\delta a}{\delta r}$, needed to evaluate eq 58, arises solely from the derivative of the exponent

$$\frac{\partial G_{\text{slim}}}{\partial a} = \frac{1}{2a} G_{\text{slim}}(r) - H_{\text{slim}}(r)$$
The integral over \( r \) in the last equation is evaluated numerically; all bracket notation terms are evaluated analytically. The density \( n_\mu \) in the auxiliary basis is computed from the atomic orbital density matrix \( P_{\mu\nu} \) which in turn is evaluated from the KS orbitals \( \phi_i \) and occupations \( f_i \).

\[
n_\mu = \sum_{\mu} C_{\mu\nu}^\mu P_{\mu\nu}
\]

(76)

\[
P_{\mu\nu} = \sum_i f_i c_\mu c_\nu
\]

(77)

\[
\phi(r) = \sum_\nu c_\nu \chi_\nu\phi(r)
\]

(78)

The above equations assume real coefficients \( c_\nu \).

### ASSOCIATED CONTENT

**Supporting Information**

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acs.jctc.1c00904.

Training and validation details for the CIDER functionals; additional benchmarking details for the Gaussian process models; linear-scaling algorithm for CIDER in Gaussian-type orbital basis sets; discussion of different possible choices for the exchange energy density; discussion of the difference between KS and HF exchange energies (PDF)

### AUTHOR INFORMATION

**Corresponding Authors**

Kyle Bystrom — John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, United States; orcid.org/0000-0003-1342-4972; Email: kylebystrom@g.harvard.edu

Boris Kozinsky — John A. Paulson School of Engineering and Applied Sciences, Harvard University, Cambridge, Massachusetts 02138, United States; orcid.org/0000-0002-0638-539X; Email: bkoz@seas.harvard.edu

Complete contact information is available at: https://pubs.acs.org/doi/10.1021/acs.jctc.1c00904

### Notes

The authors declare no competing financial interest.

### ACKNOWLEDGMENTS

The authors thank Prof. Donald Truhlar for helpful discussions regarding use of the Minnesota 2015 Database. This work was supported by the US Department of Defense MURI under Award No. N00014-20-1-2418 and by the STC Center for Integrated Quantum Materials, NSF Grant No. DMR-1231319. K.B. was supported by the Department of Defense (DoD)
through the National Defense Science & Engineering Graduate (NDSEG) Fellowship Program.

REFERENCES

(57) Vandermause, J.; Xie, Y.; Lim, J. S.; Owen, C. J.; Kozinsky, B. Active learning of reactive Bayesian force fields: Application to heterogeneous hydrogen-platinum catalysis dynamics. arXiv 2106.04481 2021 DOI: 10.21203/rs.3.rs.1178160/cv.
(70) interpolation.py Python Package; https://github.com/EconForge/interpolation.py.