AIP | The Journal of Chemical Physics

Efficient implementation of effective core potential integrals and gradients on graphical processing units

Chenchen Song, Lee-Ping Wang, Torsten Sachse, Julia Preiß, Martin Presselt, and Todd J. Martínez

Citation: The Journal of Chemical Physics **143**, 014114 (2015); doi: 10.1063/1.4922844

View online: http://dx.doi.org/10.1063/1.4922844

View Table of Contents: http://scitation.aip.org/content/aip/journal/jcp/143/1?ver=pdfcov

Published by the AIP Publishing

Efficient implementation of effective core potential integrals and gradients on graphical processing units

Chenchen Song,[1,2] Lee-Ping Wang,[1,2] Torsten Sachse,[3] Julia Preiß,[3] Martin Presselt,[3] and Todd J. Martínez[1,2]
[1]*Department of Chemistry and the PULSE Institute, Stanford University, Stanford, California 94305, USA*
[2]*SLAC National Accelerator Laboratory, Menlo Park, California 94025, USA*
[3]*Institute for Physical Chemistry, Friedrich Schiller University Jena, Helmholtzweg 4, 07743 Jena, Germany*

(Received 23 April 2015; accepted 9 June 2015; published online 7 July 2015)

Effective core potential integral and gradient evaluations are accelerated via implementation on graphical processing units (GPUs). Two simple formulas are proposed to estimate the upper bounds of the integrals, and these are used for screening. A sorting strategy is designed to balance the workload between GPU threads properly. Significant improvements in performance and reduced scaling with system size are observed when combining the screening and sorting methods, and the calculations are highly efficient for systems containing up to 10 000 basis functions. The GPU implementation preserves the precision of the calculation; the ground state Hartree-Fock energy achieves good accuracy for CdSe and ZnTe nanocrystals, and energy is well conserved in *ab initio* molecular dynamics simulations. © *2015 AIP Publishing LLC.* [http://dx.doi.org/10.1063/1.4922844]

I. INTRODUCTION

In many chemical problems of interest, the important chemical properties are mainly determined by the valence electrons, while the core electrons remain almost unchanged. The core electrons affect the potential field with which the valence electrons interact via shielding of the nucleus and the Pauli exclusion principle. As the computational cost of *ab initio* calculations increases rapidly with the number of electrons, computational complexity can be reduced if only the valence electrons are treated explicitly while the effects of the core electrons are approximated.

Hellmann[1,2] and Gombas[3] were the first to suggest replacing the interactions between valence and core electrons with an effective potential. Phillips and Kleinman[4] then provided a rigorous framework, where the valence Hamiltonian contains a pseudo-potential term and a core-valence interaction term. Since then, several attempts have been made in order to simplify the rigorous formula.[5,6] Today, the norm-conserving[7] and ultrasoft[8] pseudopotentials, along with the projector-augmented wave method,[9] are essential to plane wave density functional theory (DFT) methods, and the *ab initio* effective core potential (ECP) first proposed by Goddard,[10] then improved by Melius,[11] Kahn,[12,13] and others,[14] is widely used in DFT and quantum chemistry calculations employing Gaussian basis sets.[15] Calculations with ECPs often achieve good agreement with all-electron calculations.[16] In addition, ECPs provide a simple way to incorporate relativistic effects[17] for heavy atoms, as the most important relativistic effects are often associated with the core electrons, i.e., "scalar relativistic effects."

The applications of pseudopotentials are truly widespread in quantum chemistry[16] and condensed matter physics.[18] They include the study of structure, properties, and dynamics of systems ranging from individual metal atoms[19] and small metal clusters[20] to semiconductor nanocrystals and ferroelectric perovskites.[21] In molecular quantum chemistry, optimized ECP coefficients and corresponding Gaussian basis sets for the valence electrons have been developed for many elements in the periodic table.[22–24] ECPs have been used in conjunction with DFT in a wide variety of systems containing metal atoms[25] and verified to reproduce the heats of formation, ionization potentials, and spectroscopic properties of transition metal complexes.[26–29] Further applications include the study of redox processes[30,31] and reaction mechanisms in inorganic catalysts[32,33] and enzymes,[34] often employing *ab initio* molecular dynamics (AIMD) for sampling the thermodynamic ensemble and exploring the possible reaction pathways. Also of note is recent work showing that ECPs can be used to add dispersion corrections in DFT calculations[35,36] or to solve the "link atom" problem in hybrid quantum mechanics/molecular mechanics (QM/MM) calculations.[37–39]

The fundamental building blocks for incorporating ECPs into AIMD simulations are the ECP integrals and the corresponding gradients. As a result, simulations on larger systems for longer timescales can be enabled by accelerating computation of the ECP integrals and gradients. One approach to accelerate integral calculations is by implementation on massively multi-threaded computing architectures, such as graphical processing units (GPUs). The first practical application of GPUs in quantum chemistry was reported by Ufimtsev and Martinez.[40] Since then, GPUs have been successfully applied to various types of quantum chemistry methods by accelerating the construction of the Fock matrix[41] and its gradient,[42,43] which are the fundamental building blocks in the Hartree-Fock (HF) method and an essential component of DFT. GPUs have enabled AIMD simulations on small protein systems comprised of more than 2000 atoms from the first three rows in the periodic table.[43] Due to the differences in hardware features between traditional processors and GPUs, the calculation procedure must be carefully designed in order to efficiently utilize the GPU's computational power.

Another commonly used strategy to accelerate integral calculations is through screening, which can reduce the underlying computational scaling with respect to system size. This strategy is widely adopted for the four-center two-electron integrals arising in the HF and DFT methods. The Schwarz bound, which is derived from the Schwarz inequality, is most commonly used to identify integrals that are small enough that they can be safely neglected. It was first applied to electron repulsion integral evaluation by Whitten[44] and later introduced into direct self-consistent field (SCF) methods by Haser and Ahlrichs.[45] By estimating upper bounds for integrals and discarding the negligible ones, the scaling of computational effort (with respect to system size) for evaluating four-center two-electron integrals can often be reduced from fourth order to second order. Even though the three-center one-electron ECP integrals and gradients are much less numerous than four-center two-electron integrals, an effective screening strategy is still beneficial to avoid computation of negligible contributions.

In this paper, we describe the formulation and implementation of the ECP integrals and gradients on the GPU. The structure of this paper is as follows. First, we briefly review the general formulation of ECP integrals and the special features of the integration scheme we adopted.[46] Then, we discuss the details of the GPU implementation, especially how to screen and sort the ECP integrals in order to improve performance and reduce scaling. Finally, we analyze the efficiency and precision of the implementation using calculation results and timing data from HF single point energy calculations and AIMD simulations for a set of trial systems.

II. METHOD

A. ECP integrals and gradients

First, we briefly summarize the derivation put forth in the paper of McMurchie and Davidson.[47] The form of the effective core potential operator for an ECP center located at the origin is

$$U_{ECP}(r) = U_{L+1}(r) + \sum_{l=0}^{L}\sum_{m=-l}^{l} |S_{lm}\rangle \left[U_l(r) - U_{L+1}(r)\right]\langle S_{lm}|, \tag{1}$$

where L is the largest angular momentum orbital appearing in the core, and the angular functions $S_{lm}(\theta,\varphi)$ are the normalized real spherical harmonics. The radial potential functions $U_{L+1}(r)$ are expressed as linear combinations of primitive radial Gaussian functions,

$$U_{L+1}(r) = \sum_{k=1}^{K} d_k r^{n_k} e^{-\zeta_k r^2}. \tag{2}$$

The radial difference potentials $[U_l(r) - U_{L+1}(r)]$ are expressed in the same way with different sets of parameters d_k, n_k, and ζ_k. For simplicity, the primitive Gaussian function in the potential operator will be represented as

$$U(r;\ d_u, n, \zeta) = d_u r^n e^{-\zeta r^2} \tag{3}$$

throughout this paper, where d_u represents the contraction coefficient of the ECP primitive Gaussian function. This paper focuses on quantum chemistry methods that use Gaussian basis functions to describe the wavefunction; as such, the bra and ket functions in the ECP integral $\langle\phi_a|\, U(r)\, |\phi_b\rangle$ are basis functions centered at the atomic positions $\mathbf{A}$ and $\mathbf{B}$ and represented using primitive Cartesian Gaussian functions by

$$\phi_a(\mathbf{r}) = d_a (x - A_x)^{a_x}\left(y - A_y\right)^{a_y}(z - A_z)^{a_z} e^{-\alpha(\mathbf{r}-\mathbf{A})^2}, \tag{4}$$

$$\phi_b(\mathbf{r}) = d_b (x - B_x)^{b_x}\left(y - B_y\right)^{b_y}(z - B_z)^{b_z} e^{-\beta(\mathbf{r}-\mathbf{B})^2}, \tag{5}$$

where we employ a local coordinate system centered at the position of the ECP center for each integral. The following two types of integrals appear in ECP evaluations:

$$\chi_{ab} = \int_0^\infty r^2 U(r) \int_\Omega \phi_a(\mathbf{r})\, \phi_b(\mathbf{r})\, d\Omega dr, \tag{6}$$

$$\gamma_{ab}^l = \int_0^\infty r^2 U(r) \int_\Omega \phi_a(\mathbf{r})\, S_{lm} d\Omega \int_{\Omega'} \phi_b(\mathbf{r})\, S_{lm} d\Omega' dr. \tag{7}$$

The integral in Eq. (6) can be evaluated as

$$\begin{aligned}\chi_{ab} = 4\pi d_u d_p &\sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}(-1)^{L_a - i} A_x^{a_x-i_x} A_y^{a_y-i_y} A_z^{a_z-i_z}\\ &\times \sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}(-1)^{L_b - j} B_x^{b_x-j_x} B_y^{b_y-j_y} B_z^{b_z-j_z}\\ &\times \sum_{\lambda=0}^{i+j}\Theta_\lambda^{i_x+j_x,i_y+j_y,i_z+j_z}(\mathbf{r_P})\, Q_\lambda(2+n+i+j,\zeta,\eta,R_P),\end{aligned} \tag{8}$$

where $\eta = \alpha + \beta$, $\mathbf{P} = (\alpha\mathbf{A} + \beta\mathbf{B})/\eta$, $R_P = |\mathbf{P}|$, $\mathbf{r_P} = \mathbf{P}/R_P$, $d_p = d_a d_b \exp\left[-\alpha\beta|\mathbf{A}-\mathbf{B}|^2/\eta\right]$, and $L_a = (a_x + a_y + a_z)$, $L_b = (b_x + b_y + b_z)$, $i = (i_x + i_y + i_z)$, $j = (j_x + j_y + j_z)$. The angular factor Θ is defined as

$$\Theta_\lambda^{i,j,k}(\mathbf{r_P}) = \sum_{\mu=-\lambda}^{\lambda} S_{\lambda\mu}(\theta_P,\varphi_P)\int S_{\lambda\mu} x_n^i y_n^j z_n^k d\Omega, \tag{9}$$

where x_n, y_n, z_n are Cartesian coordinates on the unit sphere, and θ_P, φ_P are the spherical coordinates of the unit vector $\mathbf{r_p}$. The radial function is defined as

$$Q_\lambda(N,\zeta,\eta,R_P) = \int_0^\infty r^N e^{-\zeta r^2} e^{-\eta(r-R_P)^2} K_\lambda(2\eta R_P r)\, dr, \tag{10}$$

where K is the modified spherical Bessel function of the first kind weighted by an exponential factor as

$$K_\lambda(z) = M_\lambda(z)\, e^{-z}. \tag{11}$$

Similarly, the integral in Eq. (7) can be computed as

$$\begin{aligned}\gamma_{ab}^l = 16\pi^2 d_u d_a d_b &\sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}(-1)^{L_a-i}A_x^{a_x-i_x}A_y^{a_y-i_y}A_z^{a_z-i_z}\\ &\times\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}(-1)^{L_b-j}B_x^{b_x-j_x}B_y^{b_y-j_y}B_z^{b_z-j_z}\\ &\times\sum_{\lambda_1=0}^{l+i}\sum_{\lambda_2=0}^{l+j}\Omega_{l,\lambda_1\lambda_2}^{i_xi_yi_z,j_xj_yj_z}(\mathbf{r_A},\mathbf{r_B})\,T_{\lambda_1,\lambda_2}(2+n+i+j,\zeta,\alpha,R_A,\beta,R_B),\end{aligned} \tag{12}$$

where the angular factor is defined as

$$\begin{aligned}\Omega_{l,\lambda_1\lambda_2}^{i_xi_yi_z,j_xj_yj_z}(\mathbf{r_A},\mathbf{r_B}) = &\sum_{m=-l}^{l}\sum_{\mu_1=-\lambda_1}^{\lambda_1}S_{\lambda_1\mu_1}(\theta_A,\varphi_A)\\ &\times\int S_{lm}S_{\lambda_1\mu_1}x_n^{i_x}y_n^{i_y}z_n^{i_z}d\Omega\\ &\times\sum_{\mu_2=-\lambda_2}^{\lambda_2}S_{\lambda_2\mu_2}(\theta_B,\varphi_B)\\ &\times\int S_{lm}S_{\lambda_2\mu_2}x_n^{j_x}y_n^{j_y}z_n^{j_z}d\Omega\end{aligned} \tag{13}$$

and the radial function is defined as

$$\begin{aligned}&T_{\lambda_1,\lambda_2}(N,\zeta,\alpha,R_A,\beta,R_B)\\ &\quad= \int_0^\infty r^N e^{-\zeta r^2} e^{-\alpha(r-R_A)^2} e^{-\beta(r-R_B)^2} K_{\lambda_1}(2\alpha R_A r)\\ &\qquad\times K_{\lambda_2}(2\beta R_B r)\, dr,\end{aligned} \tag{14}$$

where $R_A = |\mathbf{A}|, R_B = |\mathbf{B}|$.

The analytical gradients can be evaluated by the same method as the integrals. We first apply the translational invariance relationship[48] to differentiate only the Cartesian Gaussian functions without differentiating the operator itself. The derivative of a Cartesian Gaussian function is given by linear combinations of Cartesian Gaussian functions with the same basis functions but different angular momentum quantum numbers, as differentiation raises the maximum angular momentum for a given basis function by one.

B. Screening method

Qualitatively, the radial integrals in Eqs. (10) and (14) go to zero gradually as the centers of the basis functions and the center of the ECP move away from each other. Using this intuition, we develop simple upper bound estimates for the size of ECP integrals that allow us to avoid calculating integrals that are negligibly small.

An upper bound should be (1) easy to compute, (2) always greater than the integrals, such that no integrals can be mistakenly screened out, and (3) as close to the integrals as possible in order to effectively screen out small integrals. In order to propose an upper bound formula that matches these three criteria, first note that for all $z > 0$, the function values of $K_\lambda(z)$ are restricted to the interval [0,1]. Defining

$$\kappa_\lambda = \max_{z>0} K_\lambda(z) \tag{15}$$

which decreases with increasing λ, the radial integral of Eq. (10) is bounded from above by

$$\begin{aligned}&\bar{Q}_\lambda(N,\zeta,\eta,R_P)\\ &\quad= \kappa_\lambda e^{-\frac{\zeta\eta}{\zeta+\eta}R_P^2}\int_0^\infty r^N e^{-(\zeta+\eta)\left(r-\frac{\eta}{\zeta+\eta}R_P\right)^2}dr,\end{aligned} \tag{16}$$

where the integrand (with its prefactors) is an *envelope* function that is always higher than the integrand in Eq. (10). Similarly, the radial integral of Eq. (14) is bounded by

$$\begin{aligned}&\bar{T}_{\lambda_1,\lambda_2}(N,\zeta,\alpha,R_A,\beta,R_B)\\ &\quad= \kappa_{\lambda_1}\kappa_{\lambda_2}e^{-\frac{\alpha\beta(R_A-R_B)^2}{\zeta+\alpha+\beta}-\frac{\zeta\left(\alpha R_A^2+\beta R_B^2\right)}{\zeta+\alpha+\beta}}\\ &\qquad\times\int_0^\infty r^N e^{-(\zeta+\alpha+\beta)\left(r-\frac{\alpha R_A+\beta R_B}{\zeta+\alpha+\beta}\right)^2}dr.\end{aligned} \tag{17}$$

Both Eqs. (16) and (17) involve integrals of the envelope function, which has the form

$$I(N,R_c,\rho) = \int_0^\infty r^N e^{-\rho(r-R_c)^2}dr. \tag{18}$$

The envelope function is used to determine the range of numerical integration of the radial function, which is the most computationally expensive part of evaluating the overall integral. We adopt the half-numerical ECP integrator proposed by Flores-Moreno and coworkers,[46] which uses a Gauss-Chebyshev

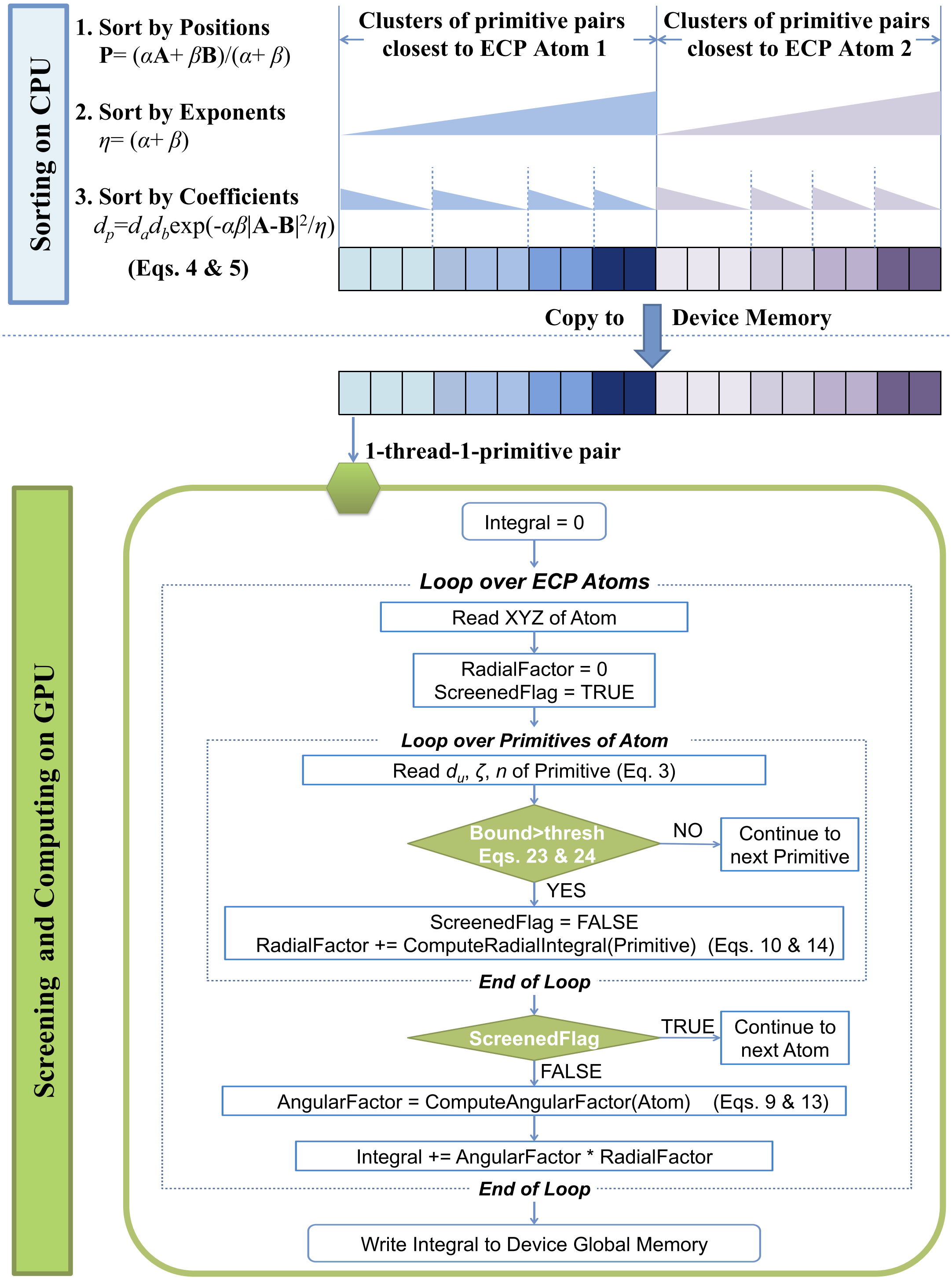
Sorting on CPU
1. Sort by Positions
P= (αA+ βB)/(α+ β)
2. Sort by Exponents
η= (α+ β)
3. Sort by Coefficients
d_p=$d_a d_b$exp(-αβ|A-B|^2/η)
(Eqs. 4 & 5)
Clusters of primitive pairs closest to ECP Atom 1
Clusters of primitive pairs closest to ECP Atom 2
Copy to Device Memory
1-thread-1-primitive pair
Screening and Computing on GPU
Integral = 0
Loop over ECP Atoms
Read XYZ of Atom
RadialFactor = 0
ScreenedFlag = TRUE
Loop over Primitives of Atom
Read d_u, ζ, n of Primitive (Eq. 3)
Bound>thresh
Eqs. 23 & 24
NO
Continue to next Primitive
YES
ScreenedFlag = FALSE
RadialFactor += ComputeRadialIntegral(Primitive) (Eqs. 10 & 14)
End of Loop
ScreenedFlag
TRUE
Continue to next Atom
FALSE
AngularFactor = ComputeAngularFactor(Atom) (Eqs. 9 & 13)
Integral += AngularFactor * RadialFactor
End of Loop
Write Integral to Device Global Memory

FIG. 1. Outline of the GPU implementation. In the CPU sorting process, primitive pairs are organized into clusters based on which ECP atom they are closest to. The wedges indicate sorting the primitive pairs within the same cluster by the absolute value of their exponents η from smallest to largest, then sorting the primitive pairs with the same exponent by the absolute value of the coefficients d_p from largest to smallest. In the GPU computing process, each thread loops over ECP atoms. If the upper bounds estimated for all primitives on the current atom are below the predefined threshold, *ScreenedFlag* is set to TRUE and calculations associated with the current atom are screened.

quadrature of the second kind[49] to generate a set of abscissas x_k in the interval (−1,1). A linear mapping is then applied as

$$2r_k = (r_{\max} - r_{\min})\, x_k + (r_{\max} + r_{\min}), \tag{19}$$

where the boundaries r_{max} and r_{min} are determined by the envelope function as

$$r_{\max} = R_c + c/\sqrt{\rho}, \tag{20}$$

$$r_{\min} = \max\left(0, R_c - c/\sqrt{\rho}\right). \tag{21}$$

The dimensionless constant c sets the boundaries of the numerical quadrature and should be chosen to cover the argument range where the integrand is nonvanishing. We chose a value of $c = 5.0$ in all of our calculations, which gives satisfactory accuracy in the calculations as can be seen below. The Gauss-Chebyshev quadrature naturally enables an adaptive quadra-

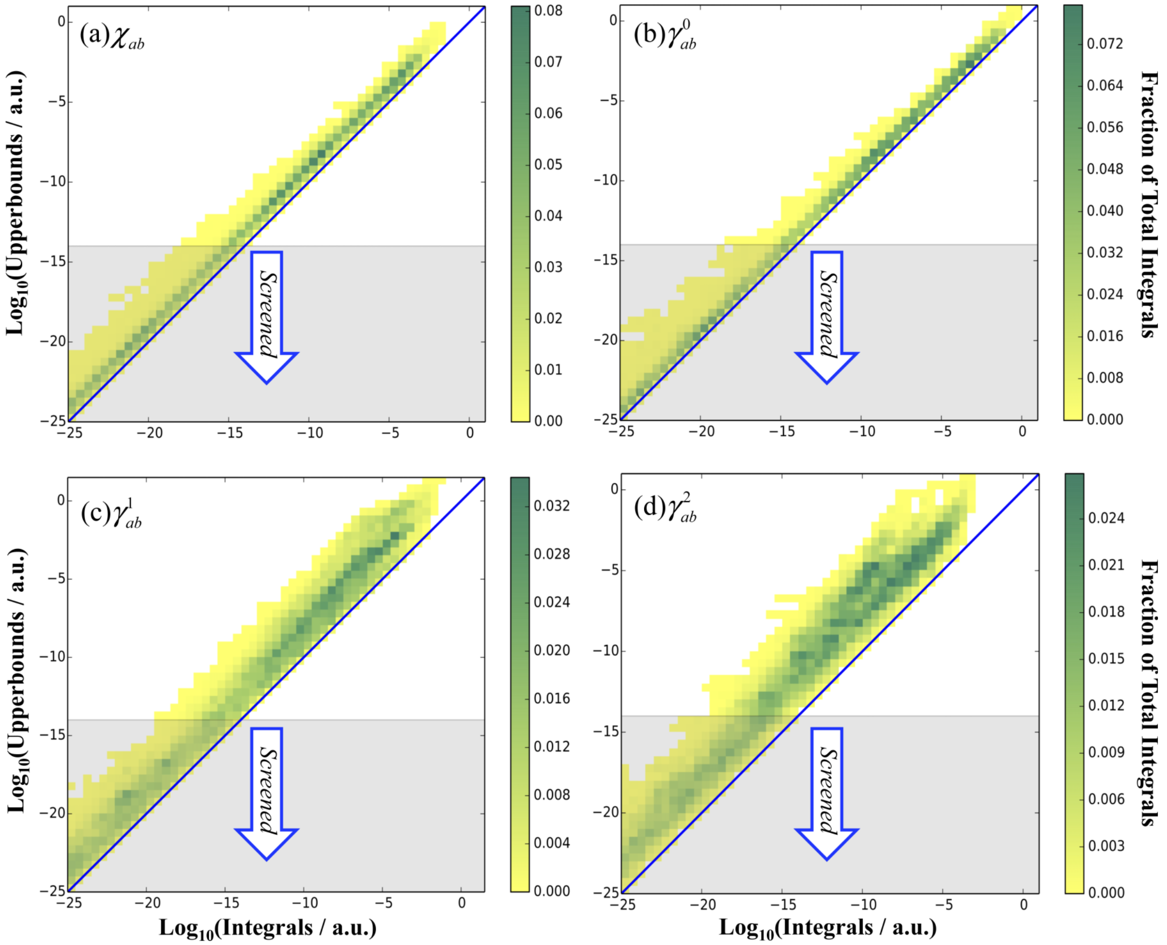

FIG. 2. Two-dimensional histograms of different types of ECP integrals comparing their magnitudes to the corresponding upper bound estimates; test systems used are Ru_{13}, $Cd_{11}Se_{11}$, and Fe_{20} with the LANL2DZ ECP and basis set. "Fraction of Total Integrals" indicates the relative number of integral/upper bound pairs in a given bin. Integrals with absolute value smaller than 1.0×10^{-25} Hartree are not shown. All populated bins are located above the diagonal line (blue), indicating that the upper bound is always greater than the absolute value of the corresponding integral. When the integral threshold is set to 1.0×10^{-14} Hartree, integrals in the grey regions are screened by the upper bound and not computed. Comparison of panels (b)-(d) shows that the bound remains effective as the ECP angular momentum increases, although it does become less tight.

ture procedure that is described in detail in Refs. 38 and 39. In our implementation, the number of quadrature points start at 7. At the beginning of each iteration, the number of quadrature points increases from p to $2p + 1$ by inserting a new quadrature point between every pair of adjacent quadrature points from the previous iteration and adding a new quadrature point on each side. Therefore, the results of the p points from the previous iteration are reused in the new iteration. The iteration terminates when convergence is reached with respect to a predefined threshold, which we set to 1.0×10^{-12} a.u. in our calculations. Generally, more quadrature points are required to reach convergence for larger integrals, but only a small number of quadrature points are necessary for small integrals.

The integrals in Eqs. (16) and (17) provide the upper bounds of the two types of radial integrals; both integrals have the form of Eq. (18), which is relatively expensive to evaluate numerically. By studying the asymptotic behavior of the integrand in the limits where $\sqrt{\rho}R_c \gg N$ or $\sqrt{\rho}R_c \ll N$, we find that

$$I(N,R_c,\rho) < \sqrt{\frac{\pi}{\rho}}\left(\frac{1}{\sqrt{\rho}}\left[\frac{1}{2\sqrt{\pi}}\Gamma\left(\frac{N+1}{2}\right)\right]^{1/N}+R_c\right)^N \tag{22}$$

which provides a simple way to estimate the contribution from the radial integrals. The derivation of Eq. (22) is given in Appendix A.

We obtained upper bounds for the angular contributions to the integral following the derivations in Appendix B, which provides the final upper bound formulas as

$$\begin{aligned}|\chi_{ab}| &< 4\pi^{3/2}|d_u d_p|\,\Xi^{L_a,L_b}(R_A+q_\chi)^{L_a}(R_B+q_\chi)^{L_b}\\ &\quad\times q_\chi^{n+2}(\zeta+\eta)^{-1/2}e^{-\frac{\zeta\eta}{\zeta+\eta}R_P^2}, \qquad (23)\\ |\gamma^l_{ab}| &< 16\pi^{5/2}|d_u d_a d_b|\,\bar{\Lambda}^{l,L_a,L_b}(R_A+q_\gamma)^{L_a}\\ &\quad\times(R_B+q_\gamma)^{L_b}q_\gamma^{n+2}(\zeta+\alpha+\beta)^{-1/2}\\ &\quad\times e^{-\frac{\alpha\beta(R_A-R_B)^2}{\zeta+\alpha+\beta}-\frac{\zeta(\alpha R_A^2+\beta R_B^2)}{\zeta+\alpha+\beta}}, \qquad (24)\end{aligned}$$

where q_χ, q_γ, Ξ^{L_a,L_b}, $\bar{\Lambda}^{l,L_a,L_b}$ are defined in Eqs. (B6), (B15), (B9), and (B16) of Appendix B, respectively. The two upper bound formulas in Eqs. (23) and (24) are simple to compute and are evaluated before calling the integrators. If the upper bound values fall below the user-defined integral threshold, then the corresponding integral can be ignored.

C. Sorting strategy and GPU implementation

As detailed in Sec. II B, due to the adaptive quadrature method and the screening method we have adopted, larger integrals are evaluated with more quadrature points than small integrals, and integrals with upper bounds below the threshold are not evaluated at all. However, the GPU hardware organizes groups of threads into *warps* (groups of 32 threads on current hardware) that are required to execute the same instructions. As a result, the primitive Gaussians of the basis functions need to be sorted properly in order to balance the workload.

The upper bound formulas in Eqs. (23) and (24) suggest the factors that influence the size of the integrals. For a given primitive Gaussian function in the ECP, Eq. (23) indicates that the size of the χ_{ab} integrals are influenced by three factors: (a) the distance from the ECP center to the "center of mass" of the primitive Gaussian pair (R_P), (b) the exponents of the primitive Gaussian pair (η), and (c) the coefficient for the primitive Gaussian pair (d_p).

The sorting strategy first groups primitive pairs by angular momentum, which is the same strategy used in our GPU implementation of Fock matrix builds.[41] Among the primitive pairs that have the same angular momentum, we implement a three level sorting strategy in order to account for the three factors listed above. Suppose there are N ECP centers in a system. First, we divide the primitive pairs into N clusters by assigning each primitive pair to the ECP center which is the closest to its center of mass, namely, the ECP center with the smallest R_p. Next, primitive pairs within the same cluster are sorted by the exponents η from smallest to largest. Finally, primitive pairs within the same cluster that also have the same exponent are sorted by their coefficients d_p from largest to smallest. In addition, when mapping the primitive pairs to the GPU threads, we require that GPU threads within a warp should always calculate primitive pairs from the same cluster. If the number of primitive pairs within a cluster is not a multiple of the warp size, then the last few threads are left idle.

For the integrals γ^l_{ab} defined by Eq. (7), based on the formulas defined in Eq. (24), a similar three-level sorting strategy is adopted, except primitive pairs are assigned to the ECP center with the smallest $(\alpha R_A + \beta R_B)/\eta$ instead of assigning primitive pairs by R_p.

Figure 1 demonstrates the outline of our GPU implementation. Each GPU thread operates on a distinct primitive pair and loops over the atom centers with ECPs. For each primitive Gaussian on the ECP, we check if the upper bound exceeds the threshold and evaluate the radial integral if so. If any radial integrals were computed, we then compute the angular factors and add the product of radial and angular integrals into the overall integral. Calculation of the integrals is easily parallelized over multiple GPUs on a single node by assigning different groups of primitive pairs to each GPU. This code has been implemented in the TeraChem quantum chemistry package.

III. RESULTS AND DISCUSSION

To test the behavior of the proposed upper bound formula, we performed computations of the ECP integrals and one-electron matrix elements on Ru_{13}, $Cd_{11}Se_{11}$, and Fe_{20}. For the systems we tested, the largest angular momentum orbital appearing in the core is $L = 2$. In each case, we compared each integral with its estimated upper bound. The distributions of the ratio of the upper bounds to the numerically exact integrals are shown in Figure 2. For all four types of integrals, the proposed upper bounds are always greater than the integrals and the upper bound is usually within a factor of 10^4 of the numerically exact integrals. Tighter bounds are likely possible, but were not found necessary for this work. The upper bounds for the integrals χ_{ab}, γ^0_{ab} appear to be more accurate (i.e., tighter bounds) than those for $\gamma^1_{ab}, \gamma^2_{ab}$, as can be seen by comparing the upper and lower panels of Figure 2. One possible reason for this behavior is that the orientation of the vectors $\mathbf{A}$ and $\mathbf{B}$ becomes more important in these integrals as l increases. Since our upper bound formula approximates the angular factors using constants $\bar{\Xi}^{L_a,L_b}$ and $\bar{\Lambda}^{l,L_a,L_b}$, its accuracy decreases as l increases.

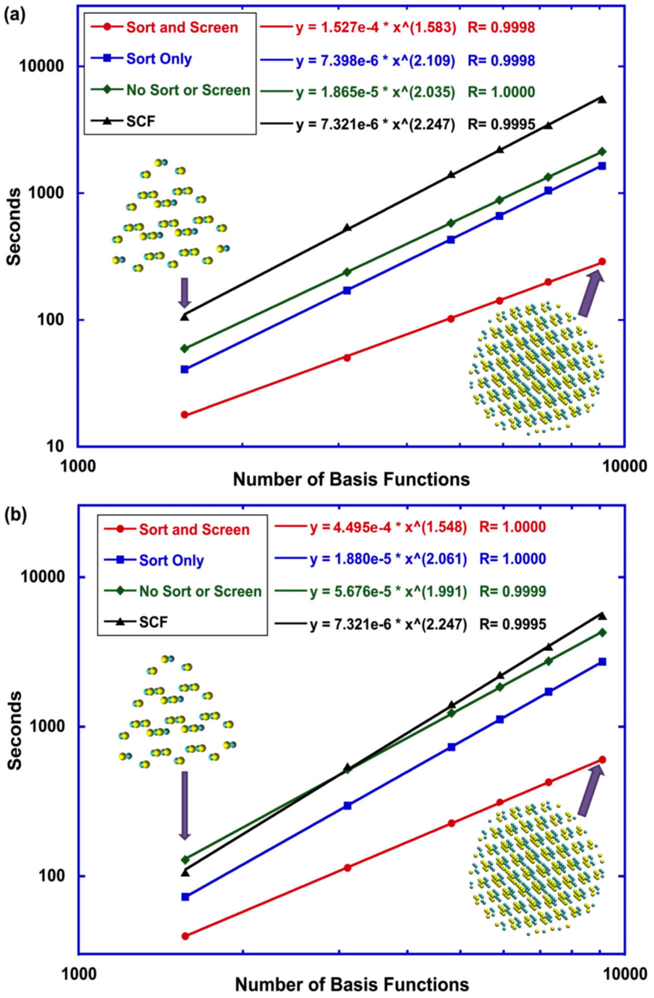

FIG. 3. Timings and scaling for computing ECP integrals (a) and gradients (b). Three different GPU implementations of ECP integrals and gradients have been tested: without sorting or screening methods (green diamonds), only sorting but no screening (blue squares), and applying both sorting and screening methods (red circles); the time for a single SCF cycle (black triangles) is provided as a reference. The ECP integral screening threshold is 1.0×10^{-13} Hartree. Data are collected from calculations on CdSe nanocrystals of increasing sizes. All calculations were performed on a quad core Intel Xeon X5570 platform using one CPU thread and one GeForce GTX Titan GPU.

We used a central processing unit (CPU) implementation of ECP integrals[13] as a reference to test the accuracy of our new GPU implementation. The difference in the Hartree-Fock energy between the CPU implementation and the new GPU implementation with screening is given in Figure S1 of the supplementary material.[50] The SCF energy differences are less than 2.5×10^{-8} Hartree for all CdSe systems and ZnTe systems tested up to 88 atoms. In addition, we compare the ECP matrix elements of $Zn_{44}Te_{44}$ between the GPU and the CPU implementations in order to verify the accuracy. As shown in Figure S2 of the supplementary material,[50] the maximum error in the matrix elements is 2×10^{-9} Hartree, and the average error is 3.6×10^{-12} Hartree.

The performance improvements due to the sorting and screening methods are shown in Figure 3. The improvements from sorting and screening are similar for integral and gradient calculations. The sorting methods primarily reduce the prefactor of the computational cost, and the screening methods effectively improve the scaling of the computational cost with respect to system size. In addition, both integral and gradient evaluations are much faster than a single SCF cycle, which involves building and diagonalizing the Fock matrix. As at least one, and often several, SCF cycles are normally required for a given timestep in *ab initio* molecular dynamics, we thus view further efficiency improvements as unnecessary at present.

Finally, we verified the consistency of the ECP integrals and gradients by carrying out *ab initio* molecular dynamics simulations on Cd_4Se_4. The initial 8-atom cluster was taken from the crystal structure of wurtzite CdSe, and initial velocities were drawn from a Maxwell-Boltzmann distribution at 1500 K. We used a velocity Verlet integrator with a 0.5 fs time step as implemented in TeraChem. As shown in Figure 4, the fluctuation in total energy is much smaller than the fluctuations in the kinetic energy and the potential energy. The drift of the total energy is 0.004 kcal/mol over 20 ps, corresponding to less than 1.0×10^{-6} kcal/mol per degree of freedom per picosecond or 0.0008% of the average kinetic energy over the simulation time; these values compare favorably with previously reported energy drifts employing similar integration methods and empirical force fields.[51] The conservation of total energy demonstrates the accuracy of both the integrals and gradients in our GPU implementation.

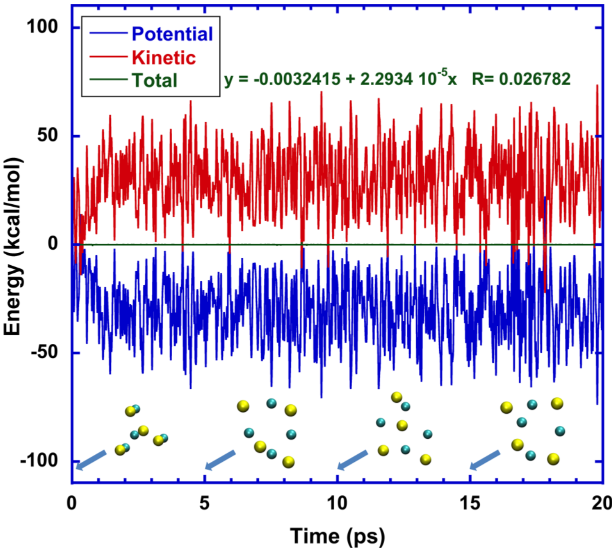

FIG. 4. Energy as a function of time in an *ab initio* molecular dynamics simulation on Cd_4Se_4 using Hartree-Fock and the LANL2DZ ECP and basis set. The initial velocities were drawn from a Maxwell-Boltzmann distribution at 1500 K. The velocity Verlet integrator with a 0.5 fs time step is used. The flat behavior of the total energy shows that the total energy is conserved, indicating consistency between the ECP integrals and gradients. The four snapshots are taken at 0 ps, 5 ps, 10 ps, and 15 ps. Se atoms are colored in yellow and Cd atoms are colored in cyan.

IV. CONCLUSIONS

By taking advantage of the properties of GPUs, we have accelerated the computation of the ECP integrals and gradients. Simple upper bound formulas were proposed which effectively screen negligible integrals and reduce the computational scaling with system size. The sorting strategy designed specifically for our GPU implementation led to efficiency improvements of more than 2× (as shown in Figure 3), primarily because this strategy balances the workload among threads.

Many interesting applications are enabled by the combination of this GPU ECP implementation and the highly efficient GPU-based Fock matrix formation engine in TeraChem. One future direction is to study heterogeneous catalytic systems, such as the methane conversion reactions through the Fischer-Tropsch synthesis,[52] usually carried out with iron catalysts. A major challenge in studying heterogeneous catalytic systems is the high number of electrons on metal clusters that tremendously increase the computational expense. As the GPU-accelerated ECP computations have greatly reduced the computational cost, it is possible to increase the size of the metal catalysts in the simulations in order to avoid edge effects and study larger systems.

Another future direction is to implement multi-centered valence electron effective potential (MC-VEEP), previously proposed by Slavíček and Martínez.[39] In this method, ECPs are used to describe the link-atoms in QM/MM simulations such that both the ground and the excited electronic states can be treated correctly. It provides a rigorous treatment of short range repulsion between the MM and QM regions while retaining a low computational cost. In addition, it can potentially be used as an *ab initio* coarse graining method. Since MC-VEEP introduces a large number of ECPs to describe the QM/MM interaction Hamiltonian, the GPU-based ECP implementation provides a route toward the practical application of this method to improve the accuracy of multi-scale simulations of large systems.

ACKNOWLEDGMENTS

This work was supported by the AMOS program within the Chemical Sciences, Geosciences, and Biosciences Division of the Office of Basic Energy Sciences, Office of Science, US Department of Energy. We thank Andrew Komornicki and Nathan Luehr for helpful suggestions and insightful discussions. Julia Preiß thanks the German academic exchange service (DAAD) and the Nagelschneider Foundation for financial support. Torsten Sachse and Martin Presselt thank the “Bundesministerium für Bildung und Forschung” (FKZ: No. 03EK3507). Torsten Sachse furthermore acknowledges financial support from the “Deutsche Bundesstiftung Umwelt.”

APPENDIX A: UPPER BOUND ESTIMATE OF ENVELOPE FUNCTION INTEGRAL

Here, we develop an estimate for the envelope function in Eq. (18),

$$I(N, R_c, \rho) = \int_0^\infty r^N e^{-\rho(r-R_c)^2} dr \tag{A1}$$

which will be used to derive the upper bound formulas for the ECP integrals in Appendix B.

We first apply a variable transformation as $t = \sqrt{\rho} r$, which gives

$$I(N, R_c, \rho) = \frac{1}{\rho^{(N+1)/2}} \int_0^\infty t^N e^{-\left(t-\sqrt{\rho}R_c\right)^2} dt. \tag{A2}$$

To get an upper bound for integral in (A2), we need to study the behavior of the following integral:

$$f(N, a) = \int_0^\infty t^N e^{-(t-a)^2} dt. \tag{A3}$$

For $N = 0$, the upper bound is trivial because

$$f(0, a) = \int_0^\infty e^{-(t-a)^2} dt < \int_{-\infty}^\infty e^{-(t-a)^2} dt = \sqrt{\pi}. \tag{A4}$$

For $N > 0$, this integral can be written in terms of the confluent hypergeometric function of the first kind ${}_1F_1$ as[53]

$$f(N, a) = \frac{1}{2} e^{-a^2} \left(aN\Gamma\left(\frac{N}{2}\right) {}_1F_1\left(\frac{N}{2}+1, \frac{3}{2}, a^2\right) + \Gamma\left(\frac{N+1}{2}\right) {}_1F_1\left(\frac{N+1}{2}, \frac{1}{2}, a^2\right)\right). \tag{A5}$$

Since ${}_1F_1$ is no simpler than evaluating the entire radial integral, we look for an alternative way to estimate the size of the integral by examining its asymptotic behavior.

Returning to the form in (A3), we make a change of variables $x = t - a$ and write out the integral as a binomial series. When a goes to infinity, we have

$$\begin{aligned}
\int_0^\infty t^N e^{(t-a)^2} dt &= \int_{-a}^\infty (x+a)^N e^{-x^2} dx \\
&= a^N \sum_{i=0}^N \binom{N}{i} a^{-i} \int_{-a}^\infty x^i e^{-x^2} dx \approx a^N \sum_{i=0}^N \binom{N}{i} a^{-i} \int_{-\infty}^\infty x^i e^{-x^2} dx \\
&= a^N \left(\int_{-\infty}^\infty e^{-x^2} dx + \frac{N(N-1)}{2a^2} \int_{-\infty}^\infty x^2 e^{-x^2} dx + O\left(a^{-4}\right)\right) \\
&= a^N \left(\sqrt{\pi} + \frac{\sqrt{\pi} N (N-1)}{4a^2} + O\left(a^{-4}\right)\right) \approx a^N \sqrt{\pi}.
\end{aligned} \tag{A6}$$

On the other hand, when a goes to zero, we have

$$\begin{aligned}
\int_0^\infty t^N e^{(t-a)^2} dt &= \int_{-a}^\infty (x+a)^N e^{-x^2} dx \\
&= \sum_{i=0}^N \binom{N}{i} \int_{-a}^\infty x^i a^{N-i} e^{-x^2} dx \approx \sum_{i=0}^N \binom{N}{i} a^{N-i} \int_0^\infty x^i e^{-x^2} dx \\
&= \int_0^\infty x^N e^{-x^2} dx + Na \int_0^\infty x^N e^{-x^2} dx + O\left(a^2\right) \\
&= \frac{1}{2}\Gamma\left(\frac{N+1}{2}\right) + \frac{Na}{2}\Gamma\left(\frac{N}{2}\right) + O\left(a^2\right) \approx \frac{1}{2}\Gamma\left(\frac{N+1}{2}\right).
\end{aligned} \tag{A7}$$

We may use the large and small a limits to devise a simple function that interpolates between the two limits and numerically show that it is larger than $f(N, a)$ for all intermediate values. One such function is the binomial series $(x + y)^N$ where x^N and y^N are set equal to the limits,

$$f(N, a) < g(N, a) = \sqrt{\pi}\left(\left(\frac{1}{2\sqrt{\pi}}\Gamma\left(\frac{N+1}{2}\right)\right)^{1/N} + a\right)^N. \tag{A8}$$

The maximum value of N is $(2 + 2L)$ for integral calculations, and $(3 + 2L)$ for gradient calculations, where L represents the highest angular momentum of the basis functions. For basis functions up to d-orbitals, the maximum N is 7. In Figure S3,[50] we compare the integrated value of $f(N, a)$ with $g(N, a)$ for N from 1 through 10.

Substituting $t = \sqrt{\rho} r$ and $a = \sqrt{\rho} R_c$ into (A8) as we have done in (A2), we now have the upper bound estimate of the envelope integral as

$$I\left(N,R_c,\rho\right) < \sqrt{\frac{\pi}{\rho}}\left(\frac{1}{\sqrt{\rho}}\left[\frac{1}{2\sqrt{\pi}}\Gamma\left(\frac{N+1}{2}\right)\right]^{1/N} + R_c\right)^{N}. \tag{A9}$$

APPENDIX B: UPPER BOUND ESTIMATES FOR ECP INTEGRALS

We first derive an easily computable formula for the upper bound of χ_{ab} which is defined in Eq. (8).

We begin by taking the absolute value of every potentially non-negative term in the summation. All the symbols and indices used are consistent with the main text. From $|a+b| \le |a| + |b|$, we have

$$\begin{aligned}
|\chi_{ab}| &= \left| \begin{aligned} &4\pi d_u d_p \sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}(-1)^{L_a-i}A_x^{a_x-i_x}A_y^{a_y-i_y}A_z^{a_z-i_z}\\ &\times\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}(-1)^{L_b-j}B_x^{b_x-j_x}B_y^{b_y-j_y}B_z^{b_z-j_z}\\ &\times\sum_{\lambda=0}^{i+j}\Theta_\lambda^{i_x+j_x,i_y+j_y,i_z+j_z}\left(\mathbf{r_P}\right)Q_\lambda\left(2+n+i+j,\zeta,\eta,R_P\right)\end{aligned}\right|\\
&\le 4\pi\left|d_u d_p\right|\sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}\left|A_x\right|^{a_x-i_x}\left|A_y\right|^{a_y-i_y}\left|A_z\right|^{a_z-i_z}\\
&\quad\times\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}\left|B_x\right|^{b_x-j_x}\left|B_y\right|^{b_y-j_y}\left|B_z\right|^{b_z-j_z}\\
&\quad\times\sum_{\lambda=0}^{i+j}\left|\Theta_\lambda^{i_x+j_x,i_y+j_y,i_z+j_z}\left(\mathbf{r_P}\right)\right|Q_\lambda\left(2+n+i+j,\zeta,\eta,R_P\right).
\end{aligned} \tag{B1}$$

Next, we may bring the components of displacement vectors, i.e., $|A_x|$, out of the sum by using $|A_w| \le R_A$, $|B_w| \le R_B$, where w represents the x, y, or z component,

$$\begin{aligned}
|\chi_{ab}| \le 4\pi\left|d_u d_p\right|&\sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}R_A^{L_a-i}\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}R_B^{L_b-j}\\
&\times\sum_{\lambda=0}^{i+j}\left|\Theta_\lambda^{i_x+j_x,i_y+j_y,i_z+j_z}(\mathbf{r_P})\right|Q_\lambda(2+n+i+j,\zeta,\eta,R_P).
\end{aligned} \tag{B2}$$

Now, we define

$$\Xi_{i,j,\lambda}^{a_xa_ya_z,b_xb_yb_z}\left(\mathbf{r_P}\right) = \sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\left\{\begin{aligned}&\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}\\ &\times\left|\Theta_\lambda^{i_x+j_x,i_y+j_y,i_z+j_z}\left(\mathbf{r_P}\right)\right|\delta_{i_x+i_y+i_z,i}\delta_{j_x+j_y+j_z,j}\end{aligned}\right\}, \tag{B3}$$

where we have introduced the Kronecker delta function. Equation (B2) can then be simplified as

$$\begin{aligned}
|\chi_{ab}| \le 4\pi\left|d_u d_p\right|&\sum_{i=0}^{L_a}\sum_{j=0}^{L_b}R_A^{L_a-i}R_B^{L_b-j}\\
&\times\sum_{\lambda=0}^{i+j}\Xi_{i,j,\lambda}^{a_xa_ya_z,b_xb_yb_z}\left(\mathbf{r_P}\right)\\
&\times Q_\lambda\left(2+n+i+j,\zeta,\eta,R_P\right).
\end{aligned} \tag{B4}$$

Applying the estimate for the radial integral defined in Eq. (A9), we get

$$\begin{aligned}
&Q_\lambda(2+n+i+j,\zeta,\eta,R_P)\\
&\quad < \kappa_\lambda\sqrt{\frac{\pi}{\zeta+\eta}}q_\chi^{2+n+i+j}e^{-\frac{\zeta\eta}{\zeta+\eta}R_P^2},
\end{aligned} \tag{B5}$$

where

$$q_\chi = \frac{1}{\sqrt{\zeta+\eta}}\tau + \frac{\eta R_P}{\zeta+\eta}, \tag{B6}$$

$$\tau = \max_{0\le L\le L_a+L_b}\left[\frac{1}{2\sqrt{\pi}}\Gamma\left(\frac{2+n+L}{2}\right)\right]^{1/(2+n+L)}. \tag{B7}$$

The value of τ can be hard coded for all possible values of $n + L_a + L_b$. Now, the expression in (B4) can be simplified as

$$
\begin{aligned}
|\chi_{ab}| < 4\pi^{\frac{3}{2}}\left|d_u d_p\right|(\zeta+\eta)^{-\frac{1}{2}}e^{-\frac{\zeta\eta}{\zeta+\eta}R_P^2} \\
\times\sum_{i=0}^{L_a}\sum_{j=0}^{L_b}R_A^{L_a-i}R_B^{L_b-j}q_\chi^{2+n+i+j} \\
\times\sum_{\lambda=0}^{i+j}\Xi_{i,j,\lambda}^{a_xa_ya_z,b_xb_yb_z}(\mathbf{r_P})\,\kappa_\lambda .
\end{aligned}
\tag{B8}
$$

Next, we search for the minimum constant $\bar{\Xi}^{L_a,L_b}$ that satisfies

$$
\sum_{\lambda=0}^{i+j}\Xi_{i,j,\lambda}^{a_xa_ya_z,b_xb_yb_z}(\mathbf{r_P})\,\kappa_\lambda \le \bar{\Xi}^{L_a,L_b}\binom{L_a}{i}\binom{L_b}{j}
\tag{B9}
$$

for any $(a_x+a_y+a_z)=L_a$, $(b_x+b_y+b_z)=L_b$, $0\le i\le L_a$, $0\le j\le L_b$ and for any unit vector $\mathbf{r_P}$. The searching procedure is straightforward. For a given (L_a,L_b), we first optimize

$$
\begin{aligned}
\tilde{\bar{\Xi}}_{i,j}^{a_xa_ya_z,b_xb_yb_z} = \max_{\mathbf{r_P}}\binom{L_a}{i}^{-1}\binom{L_b}{j}^{-1} \\
\times\sum_{\lambda=0}^{i+j}\Xi_{i,j,\lambda}^{a_xa_ya_z,b_xb_yb_z}(\mathbf{r_P})\,\kappa_\lambda
\end{aligned}
\tag{B10}
$$

for all valid $a_x, a_y, a_z, b_x, b_y, b_z, i, j$. The largest value among $\tilde{\bar{\Xi}}_{i,j}^{a_xa_ya_z,b_xb_yb_z}$ is selected as $\bar{\Xi}^{L_a,L_b}$; since the search over $\mathbf{r_P}$ is done numerically over a dense grid of points, we multiply the maximum value by a factor of 1.2 to ensure we are above any numerical noise. The value of $\bar{\Xi}^{L_a,L_b}$ can be hard coded.

Including binomial coefficients in the definition of $\bar{\Xi}^{L_a,L_b}$ allows us to remove the final two summation symbols,

$$
\begin{aligned}
|\chi_{ab}| &< 4\pi^{\frac{3}{2}}\left|d_u d_p\right|\bar{\Xi}^{L_a,L_b}\left[\sum_{i=0}^{L_a}\binom{L_a}{i}R_A^{L_a-i}q_\chi^i\right] \\
&\quad\times\left[\sum_{j=0}^{L_b}\binom{L_b}{j}R_B^{L_b-j}q_\chi^j\right]q_\chi^{2+n}(\zeta+\eta)^{-\frac{1}{2}}e^{-\frac{\zeta\eta}{\zeta+\eta}R_P^2} \\
&= 4\pi^{\frac{3}{2}}\left|d_u d_p\right|\bar{\Xi}^{L_a,L_b}\left(R_A+q_\chi\right)^{L_a}\left(R_B+q_\chi\right)^{L_b} \\
&\quad\times q_\chi^{2+n}(\zeta+\eta)^{-\frac{1}{2}}e^{-\frac{\zeta\eta}{\zeta+\eta}R_P^2}.
\end{aligned}
\tag{B11}
$$

The derivation of the upper bound function for γ_{ab}^l defined in Eq. (12) follows a similar procedure as for χ_{ab}. By applying $|a+b|\le|a|+|b|$ and $|A_w|\le R_A, |B_w|\le R_B$, we have

$$
\begin{aligned}
\left|\gamma_{ab}^l\right| \le 16\pi^2\left|d_u d_a d_b\right|\sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z} \\
\times R_A^{L_a-i}\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z}R_B^{L_b-j} \\
\times\sum_{\lambda_1=0}^{l+i}\sum_{\lambda_2=0}^{l+j}\left|\Omega_{l,\lambda_1\lambda_2}^{i_xi_yi_z,j_xj_yj_z}(\mathbf{r_A},\mathbf{r_B})\right|T_{\lambda_1,\lambda_2} \\
\times\left(2+n+i+j,\zeta,\alpha,R_A,\beta,R_B\right).
\end{aligned}
\tag{B12}
$$

Similarly, by defining

$$
\Lambda_{i,j,\lambda_1,\lambda_2}^{l,a_xa_ya_z,b_xb_yb_z}(\mathbf{r}_A,\mathbf{r_B}) = \sum_{i_x=0}^{a_x}\sum_{i_y=0}^{a_y}\sum_{i_z=0}^{a_z}\sum_{j_x=0}^{b_x}\sum_{j_y=0}^{b_y}\sum_{j_z=0}^{b_z}\left\{\begin{aligned}&\binom{a_x}{i_x}\binom{a_y}{i_y}\binom{a_z}{i_z}\binom{b_x}{j_x}\binom{b_y}{j_y}\binom{b_z}{j_z} \\ &\times\left|\Omega_{l,\lambda_1\lambda_2}^{i_xi_yi_z,j_xj_yj_z}(\mathbf{r_A},\mathbf{r_B})\right|\delta_{i_x+i_y+i_z,i}\delta_{j_x+j_y+j_z,j}\end{aligned}\right\}
\tag{B13}
$$

and applying the estimate of the radial integral, we get

$$
\begin{aligned}
\left|\gamma_{ab}^l\right| < 16\pi^{\frac{5}{2}}\left|d_u d_a d_b\right|e^{-\frac{\alpha\beta(R_A-R_B)^2}{\zeta+\alpha+\beta}-\frac{\zeta\left(\alpha R_A^2+\beta R_B^2\right)}{\zeta+\alpha+\beta}} \\
\times(\zeta+\alpha+\beta)^{-\frac{1}{2}}\sum_{i=0}^{L_a}\sum_{j=0}^{L_b}R_A^{L_a-i}R_B^{L_b-j}q_\gamma^{2+n+i+j} \\
\times\sum_{\lambda_1=0}^{l+i}\sum_{\lambda_2=0}^{l+j}\Lambda_{i,j,\lambda_1,\lambda_2}^{l,a_xa_ya_z,b_xb_yb_z}(\mathbf{r_A},\mathbf{r_B})\,\kappa_{\lambda_1}\kappa_{\lambda_2},
\end{aligned}
\tag{B14}
$$

where

$$
q_\gamma = \frac{1}{\sqrt{\zeta+\alpha+\beta}}\tau + \frac{\alpha R_A+\beta R_B}{\zeta+\alpha+\beta}
\tag{B15}
$$

the definition of τ is same as (B7).

We then search for the minimum constant $\overline{\Lambda}^{l,L_a,L_b}$ which satisfies

$$
\begin{aligned}
\sum_{\lambda_1=0}^{l+i}\sum_{\lambda_2=0}^{l+j}\Lambda_{i,j,\lambda_1,\lambda_2}^{l,a_xa_ya_z,b_xb_yb_z}(\mathbf{r_A},\mathbf{r_B})\,\kappa_{\lambda_1}\kappa_{\lambda_2} \\
\le\overline{\Lambda}^{l,L_a,L_b}\binom{L_a}{i}\binom{L_b}{j}
\end{aligned}
\tag{B16}
$$

for any $(a_x+a_y+a_z)=L_a$, $(b_x+b_y+b_z)=L_b$, $0\le i\le L_a$, $0\le j\le L_b$ and for any unit vector $\mathbf{r_A}$ and $\mathbf{r_B}$. The searching procedure is similar to the procedure in χ_{ab}.

This brings us to the final expression

$$
\begin{aligned}
\gamma_{ab}^l &< 16\pi^{\frac{5}{2}}\left|d_u d_a d_b\right|\bar{\Lambda}^{l,L_a,L_b}\left[\sum_{i=0}^{L_a}\binom{L_a}{i}R_A^{L_a-i}q_\gamma^i\right] \\
&\quad\times\left[\sum_{j=0}^{L_b}\binom{L_b}{j}R_B^{L_b-j}q_\gamma^j\right] \\
&\quad\times q_\gamma^{2+n}(\zeta+\alpha+\beta)^{-\frac{1}{2}}e^{-\frac{\alpha\beta(R_A-R_B)^2}{\zeta+\alpha+\beta}-\frac{\zeta\left(\alpha R_A^2+\beta R_B^2\right)}{\zeta+\alpha+\beta}} \\
&= 16\pi^{\frac{5}{2}}\left|d_u d_a d_b\right|\bar{\Lambda}^{l,L_a,L_b}\left(R_A+q_\gamma\right)^{L_a}\left(R_B+q_\gamma\right)^{L_b} \\
&\quad\times q_\gamma^{2+n}(\zeta+\alpha+\beta)^{-\frac{1}{2}}e^{-\frac{\alpha\beta(R_A-R_B)^2}{\zeta+\alpha+\beta}-\frac{\zeta\left(\alpha R_A^2+\beta R_B^2\right)}{\zeta+\alpha+\beta}}.
\end{aligned}
\tag{B17}
$$

[1] H. Hellmann, "A new approximation method in the problem of many electrons," J. Chem. Phys. 3, 61 (1935).

[2] H. Hellmann and W. Kassatotschkin, "Metallic binding according to the combined approximation procedure," J. Chem. Phys. **4**, 324 (1936).

[3] P. Gombas, "About the metallic bond," Z. Phys. **94**, 473 (1935).

[4] J. C. Phillips and L. Kleinman, "New method for calculating wave functions in crystals and molecules," Phys. Rev. **116**, 287 (1959).
[5] J. D. Weeks and S. A. Rice, "Use of pseudopotentials in atomic-structure calculations," J. Chem. Phys. **49**, 2741 (1968).
[6] G. Simons and A. Mazziott, "Atomic and molecular pseudopotential studies using Gaussian orbitals," J. Chem. Phys. **52**, 2449 (1970).
[7] D. R. Hamann, M. Schluter, and C. Chiang, "Norm-conserving pseudopotentials," Phys. Rev. Lett. **43**, 1494 (1979).
[8] D. Vanderbilt, "Soft self-consistent pseudopotentials in a generalized eigenvalue formalism," Phys. Rev. B **41**, 7892 (1990).
[9] P. E. Blochl, "Projector augmented-wave method," Phys. Rev. B **50**, 17953 (1994).
[10] W. A. Goddard, "New foundation for use of pseudopotentials in metals," Phys. Rev. **174**, 659 (1968).
[11] C. F. Melius and W. A. Goddard, "*Ab initio* effective potentials for use in molecular quantum-mechanics," Phys. Rev. A **10**, 1528 (1974).
[12] L. R. Kahn and W. A. Goddard, "*Ab initio* effective potentials for use in molecular calculations," J. Chem. Phys. **56**, 2685 (1972).
[13] L. R. Kahn, P. Baybutt, and D. G. Truhlar, "*Ab initio* effective core potentials - Reduction of all-electron molecular-structure calculations to calculations involving only valence-electrons," J. Chem. Phys. **65**, 3826 (1976).
[14] P. A. Christiansen, Y. S. Lee, and K. S. Pitzer, "Improved *ab initio* effective core potentials for molecular calculations," J. Chem. Phys. **71**, 4445 (1979).
[15] P. Schwerdtfeger, "The pseudopotential approximation in electronic structure theory," ChemPhysChem **12**, 3143 (2011).
[16] M. Krauss and W. J. Stevens, "Effective potentials in molecular quantum-chemistry," Ann. Rev. Phys. Chem. **35**, 357 (1984).
[17] Y. S. Lee, W. C. Ermler, and K. S. Pitzer, "*Ab initio* effective core potentials including relativistic effects. 1. Formalism and applications to Xe and Au atoms," J. Chem. Phys. **67**, 5861 (1977).
[18] W. E. Pickett, "Pseudopotential methods in condensed matter applications," Comput. Phys. Rep. **9**, 115 (1989).
[19] L. Mitas, "Quantum Monte Carlo calculation of the Fe atom," Phys. Rev. A **49**, 4411 (1994).
[20] J. L. Martins, J. Buttet, and R. Car, "Electronic and structural-properties of sodium clusters," Phys. Rev. B **31**, 1804 (1985).
[21] R. D. Kingsmith and D. Vanderbilt, "1st-principles investigation of ferroelectricity in perovskite compounds," Phys. Rev. B **49**, 5828 (1994).
[22] P. J. Hay and W. R. Wadt, "*Ab initio* effective core potentials for molecular calculations - Potentials for the transition-metal atoms Sc to Hg," J. Chem. Phys. **82**, 270 (1985).
[23] D. Andrae, U. Haussermann, M. Dolg, H. Stoll, and H. Preuss, "Energy-adjusted *ab initio* pseudopotentials for the 2nd and 3rd row transition-elements," Theor. Chim. Acta **77**, 123 (1990).
[24] W. J. Stevens, H. Basch, and M. Krauss, "Compact effective potentials and efficient shared-exponent basis-sets for the 1st-row and 2nd-row atoms," J. Chem. Phys. **81**, 6026 (1984).
[25] C. J. Cramer and D. G. Truhlar, "Density functional theory for transition metals and transition metal chemistry," Phys. Chem. Chem. Phys. **11**, 10757 (2009).
[26] Y. Yue, M. N. Weaver, and K. M. Merz, Jr., "Assessment of the "6-31+G** + LANL2DZ" mixed basis set coupled with density functional theory methods and the effective core Potential: Prediction of heats of formation and ionization potentials for first-row-transition-metal complexes," J. Phys. Chem. A **113**, 9843 (2009).
[27] N. Hebben, H.-J. Himmel, G. Eickerling, C. Herrmann, M. Reiher, V. Herz, M. Presnitz, and W. Scherer, "The electronic structure of the Tris(ethylene) complexes [M(C2H4)3] (M=Ni, Pd, and Pt): A combined experimental and theoretical study," Chem. Eur. J. **13**, 10078 (2007).
[28] J. Preiß, M. Jäger, S. Rau, B. Dietzek, J. Popp, T. Martínez, and M. Presselt, "How does peripheral functionalization of Ruthenium(II)-terpyridine complexes affect spatial charge redistribution after photoexcitation at the Franck-Condon point?," ChemPhysChem **16**, 1395 (2015).
[29] M. Wächtler, S. Kupfer, J. Guthmuller, S. Rau, L. González, and B. Dietzek, "Structural control of photoinduced dynamics in 4H-Imidazole-Ruthenium dyes," J. Phys. Chem. C **116**, 25664 (2012).
[30] M. H. Baik and R. A. Friesner, "Computing redox potentials in solution: Density functional theory as a tool for rational design of redox agents," J. Phys. Chem. A **106**, 7407 (2002).
[31] L.-P. Wang and T. Van Voorhis, "A polarizable QM/MM explicit solvent Model for computational electrochemistry in water," J. Chem. Theory. Comput. **8**, 610 (2012).
[32] L.-P. Wang and T. Van Voorhis, "Direct-coupling O-2 bond forming a pathway in cobalt oxide water oxidation catalysts," J. Phys. Chem. Lett. **2**, 2200 (2011).
[33] X. Yang and M. H. Baik, "cis,cis- (bpy)(2)(RuO)-O-V (2)O4+ catalyzes water oxidation formally via in situ generation of radicaloid Ru-IV-O center dot," J. Am. Chem. Soc. **128**, 7476 (2006).
[34] R. Wu, P. Hu, S. Wang, Z. Cao, and Y. Zhang, "Flexibility of catalytic Zinc coordination in thermolysin and HDAC8: A born-oppenheimer *ab initio* QM/MM molecular dynamics study," J. Chem. Theory Comput. **6**, 337 (2010).
[35] E. Torres and G. A. DiLabio, "A (Nearly) Universally applicable method for modelling noncovalent interactions using B3LYP," J. Phys. Chem. Lett. **3**, 1738 (2012).
[36] G. A. DiLabio and M. Koleini, "Dispersion-correcting potentials can significantly improve the bond dissociation enthalpies and noncovalent binding energies predicted by density-functional theory," J. Chem. Phys. **140**, 18A542 (2014).
[37] G. A. DiLabio, R. A. Wolkow, and E. R. Johnson, "Efficient silicon surface and cluster modeling using quantum capping potentials," J. Chem. Phys. **122**, 044708 (2005).
[38] G. A. DiLabio, M. M. Hurley, and P. A. Christiansen, "Simple one-electron quantum capping potentials for use in hybrid QM/MM studies of biological molecules," J. Chem. Phys. **116**, 9578 (2002).
[39] P. Slavíček and T. J. Martinez, "Multicentered valence electron effective potentials: A solution to the link atom problem for ground and excited electronic states," J. Chem. Phys. **124**, 084107 (2006).
[40] I. S. Ufimtsev and T. J. Martinez, "Quantum chemistry on graphical processing units. 1. Strategies for two-electron integral evaluation," J. Chem. Theory Comput. **4**, 222 (2008).
[41] I. S. Ufimtsev and T. J. Martinez, "Quantum chemistry on graphical processing units. 2. Direct self-consistent-field implementation," J. Chem. Theory Comput. **5**, 1004 (2009).
[42] I. S. Ufimtsev and T. J. Martinez, "Quantum chemistry on graphical processing units. 3. Analytical energy gradients, geometry optimization, and first principles molecular dynamics," J. Chem. Theory Comput. **5**, 2619 (2009).
[43] I. S. Ufimtsev, N. Luehr, and T. J. Martinez, "Charge transfer and polarization in solvated proteins from *ab initio* molecular dynamics," J. Phys. Chem. Lett. **2**, 1789 (2011).
[44] J. L. Whitten, "Coulombic potential-energy integrals and approximations," J. Chem. Phys. **58**, 4496 (1973).
[45] M. Haser and R. Ahlrichs, "Improvements on the direct SCF method," J. Comput. Chem. **10**, 104 (1989).
[46] R. Flores-Moreno, R. J. Alvarez-Mendez, A. Vela, and A. M. Koster, "Half-numerical evaluation of pseudopotential integrals," J. Comput. Chem. **27**, 1009 (2006).
[47] L. E. McMurchie and E. R. Davidson, "Calculation of integrals over *ab initio* pseudopotentials," J. Comput. Phys. **44**, 289 (1981).
[48] J. Breidung, W. Thiel, and A. Komornicki, "Analytical 2nd derivatives for effective core potentials," Chem. Phys. Lett. **153**, 76 (1988).
[49] J. M. Perezjorda, E. Sanfabian, and F. Moscardo, "A simple, reliable and efficient scheme for automatic numerical-integration," Comput. Phys. Commun. **70**, 271 (1992).
[50] See supplementary material at http://dx.doi.org/10.1063/1.4922844 for the absolute errors in the ground state Hartree-Fock energy and matrix elements of the GPU implementation compared against the referenced CPU implementation, and the demonstration of Eq. (1.8).
[51] J. A. Izaguirre, S. Reich, and R. D. Skeel, "Longer time steps for molecular dynamics," J. Chem. Phys. **110**, 9853 (1999).
[52] G. Xiaoguang, F. Guangzong, L. Gang, M. Hao, F. Hongjun, Y. Liang, M. Chao, W. Xing, D. Dehui, W. Mingming, T. Dali, S. Rui, Z. Shuo, L. Jiangi, S. Litao, T. Zichao, P. Xiulian, and B. Xinhe, "Direct nonoxidative conversion of methane to ethylene, aromatics, and hydrogen," Science **344**, 616 (2014).
[53] I. S. Gradshteyn and I. M. Ryzhik, *Tables of Integrals, Series and Products*, 7th ed. (Elsevier, Burlington, MA, 2007).