NONLINEAR APPROXIMATIONS FOR ELECTRONIC STRUCTURE CALCULATIONS

G. BEYLKIN AND T. S. HAUT *

ABSTRACT. We present a new method for electronic structure calculations based on novel algorithms for nonlinear approximations. We maintain a functional form for the spatial orbitals as a linear combination of products of decaying exponentials and spherical harmonics centered at the nuclear cusps. Although such representations bare some resemblance to the classical Slater-type orbitals, the complex-valued exponents in the representations are dynamically optimized via recently developed algorithms, yielding highly accurate solutions with guaranteed error bounds. These new algorithms make dynamic optimization an effective way to combine the efficiency of Slater-type orbitals with the adaptivity of modern multiresolution methods.

We develop numerical calculus suitable for electronic structure calculations. For any spatial orbital in this functional form, we represent its product with the Coulomb potential, its convolution with the Poisson kernel, etc., in the same functional form with optimized parameters. Algorithms for this purpose scale linearly in the number of nuclei. We compute electronic structure by casting the relevant equations in an integral form and solving for the spatial orbitals via iteration. As an example, for several diatomic molecules we solve the Hartree-Fock equations with speeds competitive to those of multi-resolution methods and achieve high accuracy using a small number of parameters.

1. INTRODUCTION

We present a new approach to electronic structure calculations based on recently developed algorithms for computing near optimal exponential approximations of functions [7, 8, 20, 21, 28]. We maintain a functional form for the spatial orbitals consisting of linear combinations of products of decaying exponentials and spherical harmonics centered at the nuclear cusps. While such representations are similar to the classical Slater-type orbitals, in the course of computation we optimize both the exponents and the coefficients in order to achieve an efficient representation of solutions and to obtain guaranteed error bounds. In this way, we combine the efficiency of traditional Slater-type representations with the adaptivity of current multiresolution methods.

An approach of using nonlinear algorithms to find solutions of quantum chemistry problems has its origins in seminal papers [9, 25, 30]. In these papers the authors used sums of Gaussians whose exponents and coefficients were optimized in a nonlinear fashion in order to capture the correct behavior near the nuclear cusps and the correct rate of decay. Similar approaches have been used (cf. [10], [26])

Key words and phrases. nonlinear approximations, electronic structure, numerical calculus, multiresolution methods, Hartree-Fock equations.

This research was partially supported by NSF grant DMS-100995 and DOE/ORNL grant 4000038129.

Proc R Soc A 469: 20130231, http://dx.doi.org/10.1098/rspa.2013.0231 .

to optimize the exponents in Slater-type approximations for diatomic molecules. Although such dynamic nonlinear optimization via traditional methods (e.g., Newton's method) may produce a very efficient representation, it is practical only for very small molecules. Consequently, construction of spatial orbitals is traditionally performed off-line and the resulting sets of functions are then used as a fixed basis, leading to the so-called "basis error" if the actual solution is not well approximated within the linear span of such fixed basis.

While methods using bases of optimized sets of Gaussians have revolutionized computational quantum chemistry, they generally lack the ability to control the approximation accuracy in a systematic way and to guarantee the error bounds. Indeed, selecting a basis set is often an art form that requires insight into the underlying solution and, once it is selected, the accuracy of the solution obtained using such basis is ultimately limited. The limitation on accuracy of the traditional approach have spurred the development of multiresolution methods (cf. [19, 31, 32, 33, 13]). Multiresolution methods systematically refine numerical grids (or basis functions) in the vicinity of the cusp-type singularities while using a relatively few grid points (or basis functions) elsewhere. Multiresolution methods have proven successful in efficiently computing highly accurate solutions and achieving guaranteed error bounds. However, these methods rely on approximating spatial orbitals using basis functions (e.g., piecewise polynomials in [19, 31, 32]) which do not resemble the spatial orbitals that typically arise in quantum chemistry calculations. As a result, multiresolution methods require many parameters to faithfully represent the cusps. Moreover, such local refinement schemes do not take advantage of the essential simplicity of the spatial orbitals far from the nuclei. While adaptive multiresolution methods are sufficiently fast to be used within one-particle theories of quantum chemistry [19, 31, 32, 33, 13], their use in solving the multiparticle Schrödinger equation [5, 6] is computationally costly.

In addition to multiresolution methods based on multiwavelets, let us also mention a mixed-basis method using plane-waves and atom-centered radial polynomials (cf. [27]), as well as interlocking multi-center grids (cf. [29]).

In the new approach, we use a functional form for the spatial orbitals that involves linear combinations of products of exponentials and spherical harmonics,

$$Y_l^m\left(\frac{\mathbf{r}-\mathbf{R}_j}{\|\mathbf{r}-\mathbf{R}_j\|}\right)e^{-\alpha^j_{lm,n}\|\mathbf{r}-\mathbf{R}_j\|} \quad \text{and} \quad Y_l^m\left(\frac{\mathbf{r}}{\|\mathbf{r}\|}\right)e^{-\alpha_{lm,n}\|\mathbf{r}\|}, \tag{1.1}$$

that capture the nuclear cusps at $\mathbf{R}_j$ and the far-field behavior. In contrast to standard approaches, the number and the values of the (potentially complex-valued) exponents $\alpha^j_{lm,n}$ and $\alpha_{lm,n}$ are not fixed in advance; instead, they are optimized throughout the course of the computation in order to achieve an efficient representation with a specified accuracy. For efficiency reasons, we also use an alternative representation for the spatial orbitals involving linear combinations of products of exponentials and spherical interpolating functions,

$$K_n\left(\frac{\mathbf{r}-\mathbf{R}_j}{\|\mathbf{r}-\mathbf{R}_j\|}\right)e^{-\alpha^j_{n,k}\|\mathbf{r}-\mathbf{R}_j\|} \quad \text{and} \quad K_n\left(\frac{\mathbf{r}}{\|\mathbf{r}\|}\right)e^{-\alpha_{n,k}\|\mathbf{r}\|}, \tag{1.2}$$

where K_n are interpolating functions on the unit sphere $\mathbb{S}^2$ associated with near optimal quadratures (see [4, 23, 22]). Specifically, the interpolating condition $K_n(\omega_{n'}) = \delta_{n,n'}$ holds at the spherical quadrature nodes $\omega_{n'} \in \mathbb{S}^2$ which are invariant under the icosahedral group [4]. It turns out that the interpolating functions K_n

can be expressed in terms of spherical harmonics (see Section 3.2) and, therefore, these two representations are functionally equivalent. In Section 3.2 we describe algorithms for converting between these two functional forms. We use one form or another depending on efficiency considerations as is typical in pseudo-spectral methods where the interpolating representation is more convenient for multiplying functions, while the spectral representation is more convenient for computing convolutions.

We develop a numerical calculus based on these functional representations. For example, for any two functions $\phi_1(\mathbf{r})$ and $\phi_2(\mathbf{r})$ that are already in this functional form, we develop algorithms to represent the product $\phi_1(\mathbf{r})\,\phi_2(\mathbf{r})$ and the convolution $\Delta^{-1}\phi_1(\mathbf{r})$ in the same functional form and with a small number of parameters. By casting the relevant electronic structure equations (e.g. the Hartree-Fock equations) in an integral form, we demonstrate that the functional forms in (1.1) and (1.2) can be used to solve for the ground states via iteration (using the framework developed in [18, 16, 17, 19]). The numerical calculus used within the iteration framework allows us to efficiently build up highly efficient representations for the solutions with guaranteed error bounds. It is also noteworthy that we compute the convolution $\left(\mu^2+\Delta\right)^{-1}\phi_1(\mathbf{r})$ in the spectral domain, which obviates the need to solve a large sparse matrix equation.

We consider the algorithms presented here to be of a preliminary nature, since we plan to explore a number of improvements and variants of this approach. However, our numerical experiments on diatomic molecules indicate that they are as efficient (e.g. for multiplying functions) or more efficient (e.g. for applying convolution kernels) than the corresponding operations in a multiwavelet framework. The algorithms developed here also allow us to achieve high accuracy using a small number of parameters. In particular, less than $3,000$ complex-valued parameters are required for representing the spatial orbitals for molecules of Helium Hydride HeH+ and of Lithium Hydride LiH with $\approx 5\times 10^{-7}$ absolute errors in the orbital energies; in contrast, standard multiresolution methods require several orders of magnitude more parameters for a comparable accuracy.

In Section 2, we briefly review a method for computing near optimal exponential representations [7, 8, 20, 21, 28] and some recently developed algorithms for interpolation and integration on the unit sphere $\mathbb{S}^2$ using near optimal quadratures (see [4, 23, 22]). In Section 3, we introduce basic forms for representing solutions of quantum chemistry problems. We then develop in Section 4 a numerical calculus based on the nonlinear representations (1.1) and (1.2) described in Section 3. We then use this numerical calculus in Section 5 to solve the Hartree-Fock equations for several diatomic molecules. Finally, we discuss in Section 6 some possible alternative formulations and directions for future research.

2. Preliminaries

In this section, we collect some results needed in the paper.

2.1. **Near optimal exponential representations.** Given a decaying function $f(r)$, $r\geq 0$, we use algorithms developed in [7, 8, 20, 21, 28] to construct an exponential representation,

$$\left| f(r) - \sum_{m=1}^{M} a_m e^{-\alpha_m r} \right| \leq \epsilon, \quad r\geq 0, \tag{2.1}$$

Algorithm 1 Computing exponential representations

(1) Compute M singular vectors $H\mathbf{u}_m = \sigma_m \overline{\mathbf{u}}_m$ of $H = [f_{i+j}]_{i,j=0}^{N}$, $m = 0, \ldots, M-1$, where $\sigma_M < \epsilon$
(2) Form the $M \times M$ matrix $U_3 = U_1^{\dagger} U_2$, where $U = (\mathbf{u}_0 \ldots u_{M-1})$, $U_1 = U(0 : N-1, 1 : M)$, and $U_2 = U(1 : N, 1 : M)$
(3) Compute the M eigenvalues α_j of U_3, which coincide with the exponents in (2.1)
(4) Compute the coefficients a_m in (2.1) via $\mathbf{a} = V^{\dagger}\mathbf{f}$, where V is the $2N \times M$ Vandermonde matrix , $i = 0, \ldots, M-1$ and $j = 0, \ldots, 2N-1$.

where ϵ is the desired approximation error and the exponents α_m, $\mathcal{R}e(\alpha_m) > 0$, and coefficients a_m are, in general, complex-valued. This representation uses a small number M of terms for the given approximation error ϵ, and results in highly efficient approximations. The approach to constructing such approximations has its origins in the so-called AAK theory for optimal rational approximations [1, 2, 3]. The version of the algorithm in this paper computes the exponents α_m and coefficients a_m from the $2N+1$ equispaced samples $f_n = f(Rn/(2N))$, $n = 0, \ldots, 2N$, where R and N are chosen so that $f(r)$ is sufficiently sampled and $|f(r)| < \epsilon$, $r \geq R$.

The basic steps of this construction are given below. In Algorithm 1, $X^{\dagger}$ denotes the pseudo-inverse of the matrix X, $X(m : n, :)$ denotes the sub-matrix consisting of rows m through n, and $X = (\mathbf{x}_1 \ldots \mathbf{x}_M)$ denotes the matrix consisting of the column vectors $\mathbf{x}_j$.

We have implemented a fast SVD solver for Step 1 using the randomized techniques in [11, 24, 15] and the fact that Hankel matrices can be applied in $\mathcal{O}(N \log(N))$ operations via the Fast Fourier Transform (FFT). Our implementation reduces the overall cost of Algorithm 1 to $\mathcal{O}\left(MN \log(N) + M^2 N\right)$ operations, where the (implicit) constant is small. Since typically $M \ll N$ (e.g. $M = 10$), the cost of the algorithm is essentially linear in the number of samples N. In our experience, the choice of the singular value $\sigma_M < \epsilon$ in Algorithm 1 always results in an $\mathcal{O}(\epsilon)$ error bound, as long as the function $f(r)$ is smooth and decays rapidly (which is the case for the applications here). In fact, in the experiments discussed in Section 5, choosing a fixed value of $\epsilon = 10^{-6}$ in Algorithm 1 was sufficient to achieve $\approx 5 \times 10^{-7}$ for the computed orbital energies in the Hartree-Fock examples. However, the approximation error should be checked a posteriori (using e.g. the already computed values $f_n = f(Rn/(2N))$), and a smaller singular value chosen if necessary. The connection between the accuracy ϵ and the Mth singular value σ_M in Algorithm 1 is one of the key features of AAK theory [1, 2, 3].

We use Algorithm 1 to compute exponential representations of radial functions $f(\|\mathbf{r}\|)$ that may or may not have a cusp at $\mathbf{r} = 0$. When the function $f(\|\mathbf{r}\|)$ does not have a cusp (e.g., $f(\|\mathbf{r}\|) = e^{-\|\mathbf{r}\|^2}$), Algorithm 1 produces the exponential representation to $f(\mathbf{r})$ that is effectively smooth at $\mathbf{r} = 0$. Specifically, in the Taylor expansion for small $\|\mathbf{r}\|$,

$$\sum_{m=1}^{M} a_m \exp(-\alpha_m \|\mathbf{r}\|) = \sum_{p} \frac{\|\mathbf{r}\|^p}{p!} \sum_{m=1}^{M} a_m \alpha_m^p, \tag{2.2}$$

the odd moments (approximately) vanish to sufficiently high order. We note that this property naturally arises from application of Algorithm 1 and is not imposed as an additional constraint.

Remark 2.1. Although Algorithm 1 suffices for computing exponential approximations when the approximation error ϵ is no smaller than $\approx 10^{-7}$ and $f(r)$ is smooth and decays rapidly (which is sufficient for many electronic structure calculations), computing more accurate approximations using this algorithm may require quadruple precision.

For this reason, one of the possible alternative formulations described in Section 6 involves the so-called reduction algorithm [20]. Specifically, if the function $f(r)$ is already a linear combination of N decaying exponentials, then the reduction algorithm constructs another representation of the same form, but with a smaller number $M \ll N$ of exponents. The basic idea behind this approach is that it is straightforward to construct a sub-optimal representation of a given function as a sum of decaying exponentials (i.e., a representation that contains an excessive number of terms for a desired approximation error ϵ); the reduction algorithm may then be used to compute another exponential representation, but with a smaller number of terms. The reduction algorithm requires $\mathcal{O}\left(M^2 N\right)$ operations, and is therefore essentially linear in the number of sub-optimal exponentials. Moreover, in contrast to Algorithm 1, the reduction algorithm reliably yields (near) optimal representations with approximation error ϵ as small as 10^{-14}, and has high efficiency even when $f(r)$ decays slowly or has $1/r$-type singularities.

2.2. **Integration and interpolation on the sphere.** Let us briefly recall some results from [4] on quadratures for efficient integration and interpolation on the sphere with nodes invariant under the icosahedral group.

Denoting the unit sphere in $\mathbb{R}^3$ as $\mathbb{S}^2 = \left\{\mathbf{x} \in \mathbb{R}^3 \,:\, \|\mathbf{x}\| = 1\right\}$, an orthonormal basis for $L^2\left(\mathbb{S}^2\right)$ is given by the spherical harmonics,

$$Y_l^m(\boldsymbol{\omega}) = Y_l^m(\theta,\phi) = \frac{1}{\sqrt{2\pi}} \overline{P}_l^m(\cos\theta)\, e^{im\phi},\ 0 \le |m| \le l, \quad l = 0, 1, \ldots, \tag{2.3}$$

where $\boldsymbol{\omega}$ is a unit vector, $\boldsymbol{\omega} = (\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta)$, the polar angle $\theta \in [0,\pi]$, the azimuthal angle $\phi \in [0, 2\pi)$ and $\overline{P}_l^m$ are the normalized associated Legendre functions,

$$\overline{P}_l^m(s) = (-1)^m \sqrt{\frac{(2l+1)\,(l-m)!}{2\,(l+m)!}} \frac{\left(1-s^2\right)^{m/2}}{2^l l!} \frac{d^{l+m}}{ds^{l+m}} \left(s^2-1\right)^l, \ \ |s| \le 1,$$

for $m \ge 0$ and by $\overline{P}_n^m = (-1)^{-m} \frac{(n+m)!}{(n-m)!} \overline{P}_n^{-m}$ for $m < 0$. The quadrature nodes constructed in [4] are invariant under the icosahedral group and, thus, do not concentrate excessively near the poles. These quadratures lead to a near optimal integration of functions on the sphere, i.e., the number of nodes N_L required to integrate functions in the subspace $\mathcal{P}_L$ of dimension $(L+1)^2$, where

$$\mathcal{P}_L = \bigoplus_{l=0}^{L} \mathcal{H}_l = \operatorname{span}\left\{Y_l^m(\theta,\phi)\,,\ |m| \le l,\ 0 \le l \le L\right\}, \tag{2.4}$$

is the subspace of spherical harmonics of maximum degree L, satisfy $(L+1)^2/(3N_L) \approx 1$. Following [4], let us consider the reproducing kernel for $\mathcal{P}_L$,

$$f(\boldsymbol{\omega}) = \int_{\mathbb{S}^2} K_L(\boldsymbol{\omega} \cdot \boldsymbol{\omega'}) f(\boldsymbol{\omega'}) \, d\Omega', \quad f \in \mathcal{P}_L, \tag{2.5}$$

with

$$K_L(\boldsymbol{\omega} \cdot \boldsymbol{\omega'}) = \sum_{l=0}^{L} \frac{2l+1}{4\pi} P_l(\boldsymbol{\omega} \cdot \boldsymbol{\omega'}) = \frac{L+1}{4\pi} P_L^{(1,0)}(\boldsymbol{\omega} \cdot \boldsymbol{\omega'}), \tag{2.6}$$

where P_l are the Legendre polynomials and $P_L^{(1,0)}$ is the Jacobi polynomial. Given a function $f \in \mathcal{P}_L$ and $N \geq 2L$ of its samples $f(\boldsymbol{\omega}_n^N)$ at the quadrature nodes $\boldsymbol{\omega}_n^N$, we arrive at an interpolation formula for the sphere (an analogue of Lagrange interpolation),

$$f(\boldsymbol{\omega}) = \sum_{n=1}^{N} w_n^N K_L(\boldsymbol{\omega} \cdot \boldsymbol{\omega}_n^N) f(\boldsymbol{\omega}_n^N) = \sum_{n=1}^{N} K_n^{(N,L)}(\boldsymbol{\omega}) f(\boldsymbol{\omega}_n^N), \quad f \in \mathcal{P}_L, \tag{2.7}$$

where

$$K_n^{(N,L)}(\boldsymbol{\omega}) = w_n^N K_L(\boldsymbol{\omega} \cdot \boldsymbol{\omega}_n^N), \quad w_n^N > 0.$$

Similar to the polynomial Lagrange interpolation, (2.7) is convenient for performing point-wise operations such as multiplication. In contrast, the representation using spherical harmonics is convenient for performing the convolutions with the (bound state) Helmholtz and Poisson kernels that arise in quantum chemistry.

We note that if a function of three variables in spherical coordinates $f(r, \boldsymbol{\omega}) \in \mathcal{P}_L$ for any fixed $r \geq 0$, then this function can be represented by N functions ($N \geq 2L$) of one variable, $\{f(r, \boldsymbol{\omega}_n^N)\}_{n=1}^N$.

We also make use of the addition theorem (see e.g. [14]), which states that for two unit vectors, $\boldsymbol{\omega}, \boldsymbol{\omega'} \in \mathbb{S}^2$,

$$K_L(\boldsymbol{\omega} \cdot \boldsymbol{\omega'}) = \sum_{l=0}^{L} \sum_{m=-l}^{l} Y_l^m(\boldsymbol{\omega}) Y_l^{m*}(\boldsymbol{\omega'}), \tag{2.8}$$

and, thus, the representation (2.7) can also be expressed in terms of spherical harmonics.

3. Basic forms for representing functions

3.1. **Basic form of the representation.** In our electronic structure calculations we represent functions in the form

$$f(\mathbf{r}) = f_{\mathrm{c}}(\mathbf{r}) + f_{\mathrm{s}}(\mathbf{r}), \tag{3.1}$$

where $f_{\mathrm{c}}(\mathbf{r})$ captures the behavior near the cusps and $f_{\mathrm{s}}(\mathbf{r})$ accounts for the remaining smooth part. We note that [27] also uses a decomposition into smooth and cusp terms (although the functional forms, and associated algorithms, are different).

Specifically, we represent the cusp part $f_{\mathrm{c}}(\mathbf{r})$ in the form

$$f_{\mathrm{c}}(\mathbf{r}) = \sum_{j=1}^{J} \sum_{n=1}^{N_{\mathrm{c}}} K_n^{\mathrm{c}}\left(\frac{\mathbf{r} - \mathbf{R}_j}{\|\mathbf{r} - \mathbf{R}_j\|}\right) f_n^{(j)}(\|\mathbf{r} - \mathbf{R}_j\|), \tag{3.2}$$

where $\mathbf{R}_j$, $j = 1, \ldots, J$ denote the positions of the nuclei. Here the radial component $f_n^{(j)}(r)$ satisfies

$$f_n^{(j)}(r) = \sum_k a_{k,n}^{(j)} \exp\left(-\alpha_{k,n}^{(j)} r\right), \tag{3.3}$$

and $K_n^{\mathrm{c}} \equiv K_n^{(N_{\mathrm{c}},L_{\mathrm{c}})}$ denotes the interpolating function on the sphere associated with N_{c} spherical quadrature nodes $\boldsymbol{\omega}_n^{\mathrm{c}} \equiv \boldsymbol{\omega}_n^{N_{\mathrm{c}}}$ and interpolation order L_{c} (see Section 2.2 for details). For simplicity we assume that the number of quadrature nodes, N_{c}, does not depend on the singularity location $\mathbf{R}_j$ (in the examples in Section 5, $N_{\mathrm{c}} = 12$). In contrast to representations based on Slater-type orbitals or Gaussian-type orbitals, the number and the values of (complex-valued) exponents $a_{k,n}^{(j)}$ in the radial representation (3.3) are *not* fixed in advance. Indeed, as described in Section 4, the exponents (and the coefficients) are determined dynamically throughout the course of the computation to achieve a desired level of accuracy while using a near optimally small number of terms. We also note that the J nuclei-centered terms that comprise the cusp part $f_{\mathrm{c}}(\mathbf{r})$ may overlap with one another (and, in fact, do overlap in the examples in Section 5).

Similarly, we represent the smooth part $f_{\mathrm{s}}(\mathbf{r})$ in the form

$$f_{\mathrm{s}}(\mathbf{r}) = \sum_{n=1}^{N_{\mathrm{s}}} K_n^{\mathrm{s}}\left(\frac{\mathbf{r}}{\|\mathbf{r}\|}\right) \sum_k a_{k,n} \exp\left(-\alpha_{k,n}\|\mathbf{r}\|\right), \tag{3.4}$$

where $K_n^{\mathrm{s}} \equiv K_n^{(N_{\mathrm{s}},L_{\mathrm{s}})}$ denotes the interpolating function on the sphere associated with N_{s} spherical quadrature nodes $\boldsymbol{\omega}_n^{\mathrm{s}} \equiv \boldsymbol{\omega}_n^{N_{\mathrm{s}}}$ and interpolation order L_{s}. The exponents and coefficients are again determined dynamically using Algorithm 1. We note that, because the form of the function $f_{\mathrm{s}}(\mathbf{r})$ efficiently captures the far-field behavior of solutions to the Hartree-Fock equation, we expect that N_{s} can also be taken reasonably small (in the examples in Section 5, $L_{\mathrm{s}} = 11$ and $N_{\mathrm{s}} = 192$ is large enough to achieve $\approx 5 \times 10^{-7}$ errors in the Hartree-Fock orbital energies). Also, by construction the exponential representations in (3.4) will have vanishing moments of high order, and $f_{\mathrm{s}}(\mathbf{r})$ will be effectively smooth at $\mathbf{r} = \mathbf{0}$ (see the discussion leading to equation (2.2)). Finally, we note that, from the interpolation property $K_n^{\mathrm{s}}(\omega_{n'}^{\mathrm{s}}) = \delta_{n,n'}$,

$$f_{\mathrm{s}}(r\omega_n^{\mathrm{s}}) = \sum_k a_{k,n} \exp\left(-\alpha_{k,n}\|\mathbf{r}\|\right).$$

This simple observation leads to a fast evaluation of $f_{\mathrm{s}}(\mathbf{r})$ on spherical grids $r_j\omega_n^{\mathrm{s}}$, and is crucial for the efficiency of the algorithms in Section 4.

As described in Section 3.2, the cusp part $f_{\mathrm{c}}(\mathbf{r})$ may be equivalently expressed in terms of spherical harmonics,

$$f_{\mathrm{c}}(\mathbf{r}) = \sum_{j=1}^{J} \sum_{l=0}^{L_{\mathrm{c}}} \sum_{m=-l}^{l} Y_l^m\left(\frac{\mathbf{r}-\mathbf{R}_j}{\|\mathbf{r}-\mathbf{R}_j\|}\right) f_{lm}^{(j)}\left(\|\mathbf{r}-\mathbf{R}_j\|\right), \tag{3.5}$$

where the radial component $f_{lm}^{(j)}(r)$ is again represented in terms of exponentials,

$$f_{lm}^{(j)}(r) = \sum_k a_{k,lm}^{(j)} \exp\left(-\alpha_{k,lm}^{(j)} r\right).$$

In an analogous way, the smooth part $f_{\mathrm{s}}(\mathbf{r})$ can also be expressed in terms of spherical harmonics,

$$f_{\mathrm{s}}(\mathbf{r}) = \sum_{l=0}^{L_{\mathrm{s}}} \sum_{m=-l}^{l} Y_l^m(\boldsymbol{\omega}) \sum_k a_{k,lm} \exp\left(-\alpha_{k,lm} \|\mathbf{r}\|\right). \tag{3.6}$$

The representations (3.2) and (3.4) based on spherical interpolating functions are convenient for algorithms such as multiplication (see Sections 4.1 and 4.2), while the forms (3.5) and (3.6) are convenient for applying convolution operators (see Section 4.3).

3.2. **Converting between interpolating and spherical representations.** We make use of two basic forms for representing $f_{\mathrm{c}}(\mathbf{r})$ and $f_{\mathrm{s}}(\mathbf{r})$ and now show how to convert between these two representations. It suffices to consider

$$f(\mathbf{r}) = \sum_{l=0}^{L} \sum_{m=-l}^{l} Y_l^m(\boldsymbol{\omega}) f_{lm}(r), \quad f_{lm}(r) = \sum_k a_{k,lm} \exp\left(-\alpha_{k,lm} r\right), \tag{3.7}$$

and

$$f(\mathbf{r}) = \sum_{n=1}^{N} K_n^{(N,L)}(\boldsymbol{\omega}) f_n(r), \quad f_n(r) = \sum_k a_{k,n} \exp\left(-\alpha_{k,n} r\right). \tag{3.8}$$

We make use of the fact that the interpolating function $K_n^{(N,L)}(\boldsymbol{\omega})$ can be written in the form

$$K_n^{(N,L)}(\boldsymbol{\omega}) = w_n^N \sum_{l=0}^{L} \sum_{m=-l}^{l} Y_l^{m,*}\left(\boldsymbol{\omega}_n^N\right) Y_l^m(\boldsymbol{\omega}), \tag{3.9}$$

where $\boldsymbol{\omega}_n^N$ and w_n^N denote the nth node and weight associated with the N-point quadrature rule (see Section 2.2).

In order to convert from (3.7) to (3.8), we have

$$\begin{aligned} f_{lm}(r) &= \int_{\mathbb{S}^2} \left(\sum_{n=1}^{N} K_n^{(N,L)}(\boldsymbol{\omega}) f_n(r) \right) Y_l^{m,*}(\boldsymbol{\omega}) \, d\Omega \\ &= \sum_{n=1}^{N} \left(\int_{\mathbb{S}^2} K_n^{(N,L)}(\boldsymbol{\omega}) Y_l^{m,*}(\boldsymbol{\omega}) \, d\Omega \right) f_n(r) \\ &= \sum_{n=1}^{N} w_n^N Y_l^{m,*}\left(\boldsymbol{\omega}_n^N\right) f_n(r), \end{aligned}$$

where the last equality uses formula (3.9). Thus, to convert from a spherical harmonic representation to an interpolating representation, it suffices to use Algorithm 1 and the above formula for evaluating $f_{lm}(r)$.

Similarly, to convert from (3.8) to (3.7), we use

$$f_n(r) = f\left(r\boldsymbol{\omega}_n^N\right) = \sum_{l=0}^{L} \sum_{m=-l}^{l} Y_l^m\left(\boldsymbol{\omega}_n^N\right) f_{lm}(r).$$

Hence, it suffices to use Algorithm 1 and the above formula for evaluating $f_n(r)$.

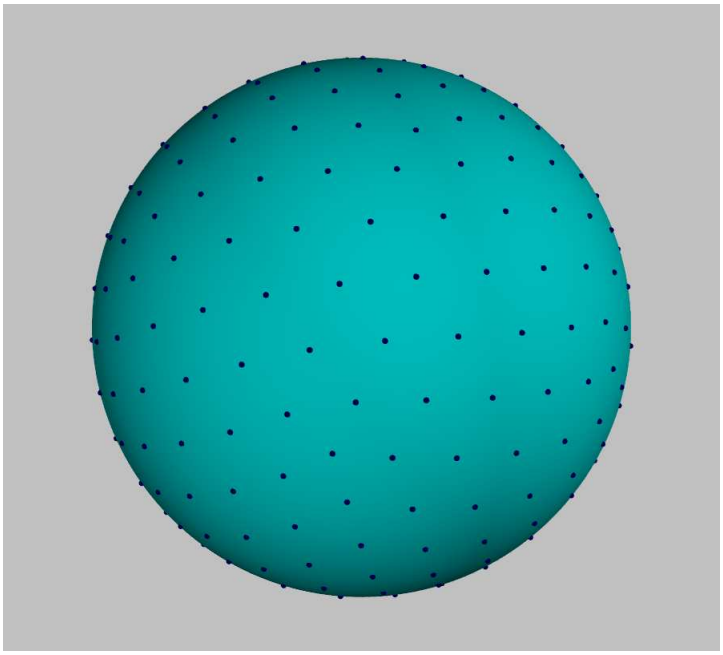


FIGURE 3.1. Sphere with 312 quadrature nodes invariant under the icosahedral group. The quadrature with these nodes and corresponding positive weights allows us to integrate exactly all functions in the subspace $\mathcal{P}_{28}$ and interpolate all functions in the subspace $\mathcal{P}_{14}$ using (2.7).

Remark 3.1. Algorithm 1 for representing $f_{lm}(r)$ as a sum of near optimal exponentials requires computing a spherical harmonic transform, for each radial grid point $r = r_j$. This can, in principle, be accomplished using a fast algorithm. While we plan to address this issue elsewhere, in the examples in Section 5 the subspace dimension L and the number N of spherical quadrature nodes (necessary to achieve high accuracy) is small, e.g., $L = 11$ and $N = 192$, and the direct evaluation is sufficiently fast. We also note that the spherical transforms (one for each radial grid point) can be computed in a trivially parallel manner.

4. Algorithms for quantum chemistry applications

We now discuss algorithms for performing electronic structure calculations using the representations from Section 3.1. For the applications that we consider, it is sufficient to develop algorithms for computing representations of $f_1(\mathbf{r}) f_2(\mathbf{r})$, $f_1(\mathbf{r}) + f_2(\mathbf{r})$, $V(\mathbf{r}) f_1(\mathbf{r})$, $\left(-\Delta + \mu^2\right)^{-1} f_1(\mathbf{r})$ $(\mu \geq 0)$, and $\Delta f_1(\mathbf{r})$ in the form discussed in Section 3.1. Here $V(\mathbf{r})$ denotes the Coulomb potential,

$$V(\mathbf{r}) = \sum_{j=1}^{J} \frac{Z_j}{\|\mathbf{r} - \mathbf{R}_j\|}. \tag{4.1}$$

We develop algorithms for the nontrivial operations, $f_1(\mathbf{r}) f_2(\mathbf{r})$, $V(\mathbf{r}) f_1(\mathbf{r})$ and $\left(-\Delta + \mu^2\right)^{-1} f_1(\mathbf{r})$, and $\Delta f_1(\mathbf{r})$.

4.1. **Multiplication of representations.** Let us consider two functions $f(\mathbf{r}) = f_{\mathrm{c}}(\mathbf{r}) + f_{\mathrm{s}}(\mathbf{r})$ and $g(\mathbf{r}) = g_{\mathrm{c}}(\mathbf{r}) + g_{\mathrm{s}}(\mathbf{r})$ in the form (3.1) in Section 3.1. We present an algorithm to construct the same type of representation for the product,

$$f(\mathbf{r})\, g(\mathbf{r}) = h_{\mathrm{c}}(\mathbf{r}) + h_{\mathrm{s}}(\mathbf{r}) + \mathcal{O}(\epsilon),$$

where ϵ is the desired approximation error.

The basic idea behind the algorithm is that, in a neighborhood of the singularity at $\mathbf{R}_j$, the product $f(\mathbf{r})\, g(\mathbf{r})$ has the form

$$f(r\boldsymbol{\omega} + \mathbf{R}_j)\, g(r\boldsymbol{\omega} + \mathbf{R}_j) = \sum_{l=0}^{L_{\mathrm{c}}} r^l \sum_{m=-l}^{l} q_{lm}^{(j)}(r)\, Y_{lm}(\boldsymbol{\omega}) + r^{L_{\mathrm{c}}+1} h_{L_{\mathrm{c}}}(r\boldsymbol{\omega}),$$

where $q_{lm}^{(j)}(r)$ is a polynomial and $h_{L_{\mathrm{c}}}(r\boldsymbol{\omega})$ is continuous. Therefore, the difference

$$f(r\boldsymbol{\omega} + \mathbf{R}_j)\, g(r\boldsymbol{\omega} + \mathbf{R}_j) - \sum_{l=0}^{L_{\mathrm{c}}} r^l \sum_{m=-l}^{l} q_{lm}^{(j)}(r)\, Y_{lm}(\boldsymbol{\omega}) = \mathcal{O}\left(r^{L_{\mathrm{c}}+1}\right)$$

has $L_{\mathrm{c}} + 1$ derivatives at $\mathbf{r} = \mathbf{R}_j$, and can be efficiently represented in the form (3.6); we also observe that the far-field behavior of $f(\mathbf{r})\, g(\mathbf{r})$ is efficiently captured by (3.6). Noting the equivalence of the spherical harmonic and interpolating representations (3.2) and (3.5), it is sufficient to construct functions $h_j(\mathbf{r})$, $j = 1, \ldots, J$,

$$h_j(\mathbf{r}) = \sum_{n=1}^{N_{\mathrm{c}}} K_n^{\mathrm{c}}\left(\frac{\mathbf{r} - \mathbf{R}_j}{\|\mathbf{r} - \mathbf{R}_j\|}\right) \sum_k c_{k,n}^{(j)} \exp\left(-\gamma_{k,n}^{(j)} \|\mathbf{r} - \mathbf{R}_j\|\right),$$

such that for $\mathbf{r}$ close to $\mathbf{R}_j$,

$$f(\mathbf{r})\, g(\mathbf{r}) - h_j(\mathbf{r}) = \mathcal{O}(\epsilon).$$

It then holds that the function $h_{\mathrm{s}}(\mathbf{r})$,

$$h_{\mathrm{s}}(\mathbf{r}) = f(\mathbf{r})\, g(\mathbf{r}) - \sum_{j=1}^{J} h_j(\mathbf{r}),$$

is effectively smooth (neglecting terms of size $\mathcal{O}(\epsilon)$), and can be efficiently represented in the form

$$h_{\mathrm{s}}(\mathbf{r}) = \sum_{n=1}^{N_{\mathrm{s}}} K_n^{\mathrm{s}}\left(\frac{\mathbf{r}}{\|\mathbf{r}\|}\right) \sum_k c_{k,n} \exp\left(-\gamma_{k,n} \|\mathbf{r}\|\right). \tag{4.2}$$

In fact, for $\mathbf{r}$ in a neighborhood of a possible cusp location $\mathbf{R}_j$,

$$h_{\mathrm{s}}(\mathbf{r}) = f(\mathbf{r})\, g(\mathbf{r}) - h_j(\mathbf{r}) + \sum_{k \neq j} h_k(\mathbf{r}) = \mathcal{O}(\epsilon) + \sum_{k \neq j} h_k(\mathbf{r}).$$

Since the functions $h_k(\mathbf{r})$, $k \neq j$, are smooth at $\mathbf{R}_j$, so is $h_{\mathrm{s}}(\mathbf{r})$ (neglecting terms of size $\mathcal{O}(\epsilon)$). We observe that choosing $L_{\mathrm{c}} = 2$ and $N_{\mathrm{c}} = 12$ yields an accuracy of $\approx 10^{-7}$ for examples in Section 5, and requires about 200 (complex-valued) parameters for each cusp.

In order to construct functions $h_j(\mathbf{r})$, we compute, via Algorithm 1, exponential representations for each spherical quadrature node $\boldsymbol{\omega}_n^{\mathrm{c}}$, $n = 1, 2, \ldots, N_{\mathrm{c}}$,

$$f(r\boldsymbol{\omega}_n^{\mathrm{c}} + \mathbf{R}_j)\, g(r\boldsymbol{\omega}_n^{\mathrm{c}} + \mathbf{R}_j) = \sum_k c_{k,n}^{(j)} \exp\left(-\gamma_{k,n}^{(j)} r\right) + \mathcal{O}(\epsilon), \quad 0 \leq r \leq R. \tag{4.3}$$

Here we can choose the cutoff R to be small, e.g. $R = \min_{k \neq j} \|\mathbf{R}_j - \mathbf{R}_k\|/5$ (only local behavior matters when subtracting off the singularity). Using interpolation on the sphere (see Section 2.2), we then have that for r small enough and all $\boldsymbol{\omega} \in \mathbb{S}^2$,

$$f(r\boldsymbol{\omega} + \mathbf{R}_j)\, g(r\boldsymbol{\omega} + \mathbf{R}_j) = \sum_{n=1}^{N_{\mathrm{c}}} K_n^{\mathrm{c}}(\boldsymbol{\omega}) \sum_k c_{k,n}^{(j)} \exp\left(-\gamma_{k,n}^{(j)} r\right) + \mathcal{O}(\epsilon), \tag{4.4}$$

The computation of $h_{\mathrm{s}}(\mathbf{r})$ is similar. In particular, for each spherical quadrature node $\boldsymbol{\omega}_n^{\mathrm{s}}$, $n = 1, 2, \ldots, N_{\mathrm{s}}$, we construct via Algorithm 1 exponential representations

$$f(r\boldsymbol{\omega}_n^{\mathrm{s}})\, g(r\boldsymbol{\omega}_n^{\mathrm{s}}) - \sum_{j=1}^{J} h_j(r\boldsymbol{\omega}_n^{\mathrm{s}}) = \sum_k c_{k,n} \exp(-\gamma_{k,n} r) + \mathcal{O}(\epsilon). \tag{4.5}$$

It then holds that, for all $r \geq 0$ and $\boldsymbol{\omega} \in \mathbb{S}^2$,

$$f(r\boldsymbol{\omega})\, g(r\boldsymbol{\omega}) - \sum_{j=1}^{J} h_j(r\boldsymbol{\omega}) = \sum_{n=1}^{N_{\mathrm{s}}} K_n^{\mathrm{s}}(\boldsymbol{\omega}) \sum_k c_{k,n} \exp(-\gamma_{k,n} r) + \mathcal{O}(\epsilon),$$

where we assume that the number N_{s} of nodes on the sphere is chosen large enough to sufficiently sample the smooth part. In using Algorithm 1, we choose the cutoff R so that the product $f(r\boldsymbol{\omega}_n^{\mathrm{s}})\, g(r\boldsymbol{\omega}_n^{\mathrm{s}})$ is smaller than the desired approximation error ϵ.

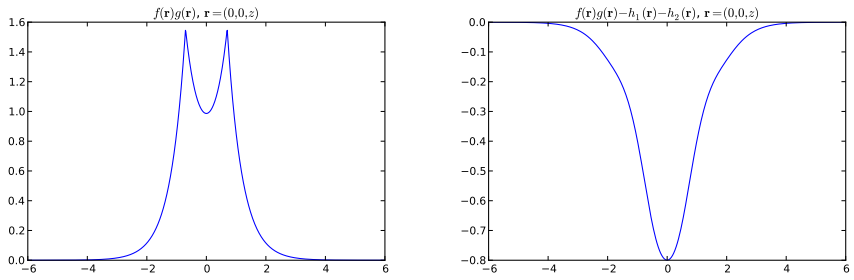
Let us now comment on the efficiency of this approach. First, constructing the cusp part, $h_{\mathrm{c}}(\mathbf{r})$, of the product $f(\mathbf{r})\, g(\mathbf{r})$ requires sampling the functions $f(r\boldsymbol{\omega}_n^{\mathrm{c}} + \mathbf{R}_j)$ and $g(r\boldsymbol{\omega}_n^{\mathrm{c}} + \mathbf{R}_j)$ on an equispaced grid for each spherical quadrature node $\boldsymbol{\omega}_n^{\mathrm{c}}$. Since the number N_{c} of spherical quadrature nodes $\boldsymbol{\omega}_n^{\mathrm{c}}$ is small (e.g., $N_{\mathrm{c}} = 12$), constructing these samples is inexpensive. In addition, the exponential representations (4.3) need only be accurate for $0 \leq r \leq R$, where R small (e.g., $R = \min_{k \neq j} \|\mathbf{R}_j - \mathbf{R}_k\|/5$) and, hence, the number of needed radial samples is also small. Once these samples are computed, applying Algorithm 1 is inexpensive, since its cost is essentially linear in the number of samples.

Similarly, constructing the smooth part, $h_{\mathrm{s}}(\mathbf{r})$, of $f(\mathbf{r})\, g(\mathbf{r})$ requires sampling the functions $f(r\boldsymbol{\omega}_n^{\mathrm{s}})$ and $g(r\boldsymbol{\omega}_n^{\mathrm{s}})$ on an equispaced grid for each spherical quadrature node $\boldsymbol{\omega}_n^{\mathrm{s}}$. Here the number, N_{s}, of spherical quadrature nodes $\boldsymbol{\omega}_n^{\mathrm{s}}$ is typically a factor of ten or so larger than N_{c} (e.g., $N_{\mathrm{s}} = 192$ in the experiments described in Section 5), and it may first appear that evaluating these samples would be more costly. However, since the smooth terms $f_{\mathrm{s}}(\mathbf{r})$, $g_{\mathrm{s}}(\mathbf{r})$, and $h_{\mathrm{s}}(\mathbf{r})$ of the multiplicands and the product share the same quadrature nodes $\boldsymbol{\omega}_n^{\mathrm{s}}$, sampling these terms requires no interpolation. In particular, recall from Section 3.1 that $f(\mathbf{r})$ is of the form $f(\mathbf{r}) = f_{\mathrm{c}}(\mathbf{r}) + f_{\mathrm{s}}(\mathbf{r})$. Since $f_{\mathrm{c}}(\mathbf{r})$ involves only N_{c} terms, where N_{c} is small, evaluating $f_{\mathrm{c}}(r\boldsymbol{\omega}_n^{\mathrm{s}})$ is inexpensive. Also, although $f_{\mathrm{s}}(\mathbf{r})$ nominally involves a sum of N_{s} terms, we can simply use the interpolation relationship,

$$f_{\mathrm{s}}(r\boldsymbol{\omega}_n^{\mathrm{s}}) = \sum_k a_{k,n} \exp(-\alpha_{k,n} \|\mathbf{r}\|),$$

so that $f_{\mathrm{s}}(\mathbf{r})$ is also inexpensive to sample.

In addition to the above considerations, we also note that the computation of the exponential representations in (4.4) and (4.5) can be done in a trivially parallel manner for each spherical quadrature node $\boldsymbol{\omega}_n^{\mathrm{c}}$ and $\boldsymbol{\omega}_n^{\mathrm{s}}$.



(a) (b)

FIGURE 4.1. The product $f(\mathbf{r})\,g(\mathbf{r})$ (a), where $f(\mathbf{r}) = g(\mathbf{r}) = e^{-\|\mathbf{r}-\mathbf{R}_1\|} + e^{-\|\mathbf{r}-\mathbf{R}_2\|}$, and the corresponding smooth part $f(\mathbf{r})\,g(\mathbf{r}) - h_1(\mathbf{r}) - h_2(\mathbf{r})$ (b) on the line $\mathbf{r} = (0,0,z)$ connecting $\mathbf{R}_1$ and $\mathbf{R}_2$.

To illustrate the basic idea of algorithm, we consider the function

$$f(\mathbf{r}) = g(\mathbf{r}) = e^{-\|\mathbf{r}-\mathbf{R}_1\|} + e^{-\|\mathbf{r}-\mathbf{R}_2\|},$$

where $\mathbf{R}_1 = (0,0,-.7)$ and $\mathbf{R}_1 = (0,0,.7)$. We use a subspace dimension $L_{\mathrm{c}} = 2$ and $N_{\mathrm{c}} = 12$ spherical quadrature nodes for the cusp part, and an error tolerance of $\epsilon = 10^{-6}$ in Algorithm 1. For the smooth part we use $L_{\mathrm{s}} = 11$ and $N_{\mathrm{s}} = 192$. In Figure 4.1a, we display the function $f(\mathbf{r})\,g(\mathbf{r})$ on the line $\mathbf{r} = (0,0,z)$ connecting the two cusp locations $\mathbf{R}_1$ and $\mathbf{R}_1$. In Figure 4.1b, we display the function $f(\mathbf{r})\,g(\mathbf{r}) - h_1(\mathbf{r}) - h_2(\mathbf{r})$ after the cusp part is subtracted. Note that we allow the two cusp terms $h_1(\mathbf{r})$ and $h_2(\mathbf{r})$ to overlap in order to avoid (artificially) sharp gradients in the representation.

4.2. **Multiplication with the potential.** Given the function $f(\mathbf{r}) = f_{\mathrm{c}}(\mathbf{r}) + f_{\mathrm{s}}(\mathbf{r})$, we now discuss how to construct a representation

$$V(\mathbf{r})\,f(\mathbf{r}) = g_{\mathrm{c}}(\mathbf{r}) + g_{\mathrm{s}}(\mathbf{r}) + \left(\sum_{j=1}^{J} \frac{v_j(\mathbf{r})}{\|\mathbf{r}-\mathbf{R}_j\|}\right)\mathcal{O}(\epsilon),$$

where $V(\mathbf{r})$ denotes the Coulomb potential (4.1) and the functions $v_j(\mathbf{r})$ decay exponentially fast. The algorithm is nearly identical to the multiplication algorithm in Section 4.1. The main difference is that the cusp part $g_{\mathrm{c}}(\mathbf{r})$ is represented in the form

$$g_{\mathrm{c}}(\mathbf{r}) = \sum_{j=1}^{J}\sum_{n=1}^{N_{\mathrm{c}}} K_n^{\mathrm{c}}\left(\frac{\mathbf{r}-\mathbf{R}_j}{\|\mathbf{r}-\mathbf{R}_j\|}\right) g_n^{(j)}\left(\|\mathbf{r}-\mathbf{R}_j\|\right),$$

where the radial component $g_n^{(j)}(r)$ is given by

$$g_j(\mathbf{r}) = \frac{1}{r}\sum_k a_{k,n}^{(j)} \exp\left(-\alpha_{k,n}^{(j)} r\right) + \sum_n b_{k,n}^{(j)} \exp\left(-\beta_{k,n}^{(j)} r\right). \tag{4.6}$$

The smooth part $g_{\mathrm{s}}(\mathbf{r})$ is again represented in the form

$$g_s(\mathbf{r}) = \sum_{n=1}^{N_{\mathrm{s}}} K_n^{\mathrm{s}}\left(\frac{\mathbf{r}}{\|\mathbf{r}\|}\right)\sum_k a_{k,n}\exp\left(-\alpha_{k,n}\|\mathbf{r}\|\right).$$

The advantage of maintaining this intermediate form is twofold. First, by incorporating the $1/r$-type singularities explicitly in the representation of $g_{\mathrm{c}}(\mathbf{r})$, only a small number of parameters need to be maintained for an accurate approximation of $V(\mathbf{r}) f(\mathbf{r})$. Second, multiplication by the potential $V(\mathbf{r})$ is always followed by an application of the Green's function, $\left(-\Delta+\mu^2\right)^{-1}(V(\mathbf{r}) f(\mathbf{r}))$, and the resulting (less singular) function can again be efficiently represented in the basic form discussed in Section 3.1 (see Section 4.3 for additional details). We note that, as it was observed in algorithms [19, 31, 32], multiplication by the potential typically creates a large number of fine scales which are then immediately reduced by convolution, with some loss of the overall efficiency. By maintaining the above intermediate form, we avoid this issue.

Similar to Section 4.1, the basic idea is that, in a neighborhood of the singularity $\mathbf{R}_j$, the product $V(\mathbf{r}) f(\mathbf{r})$ has the form

$$\begin{aligned} V(r\boldsymbol{\omega}+\mathbf{R}_j) f(r\boldsymbol{\omega}+\mathbf{R}_j) &= \frac{Z_j}{r} f(r\boldsymbol{\omega}+\mathbf{R}_j) + \\ &\quad \left(\sum_{k\neq j}^{J} \frac{Z_k}{\|r\boldsymbol{\omega}+\mathbf{R}_j-\mathbf{R}_k\|}\right) f(r\boldsymbol{\omega}+\mathbf{R}_j) \\ &= \sum_{l=0}^{L_{\mathrm{c}}} r^{l-1} \sum_{m=-l}^{l} q_{lm}^{(j)}(r) Y_{lm}(\boldsymbol{\omega}) + r^{L_{\mathrm{c}}} h_{L_{\mathrm{c}}}(r\boldsymbol{\omega}), \end{aligned}$$

where $q_{lm}^{(j)}(r)$ is a polynomial and $h_{L+1}(r\boldsymbol{\omega})$ is continuous. Therefore, the difference

$$f(r\boldsymbol{\omega}+\mathbf{R}_j) g(r\boldsymbol{\omega}+\mathbf{R}_j) - \sum_{l=0}^{L_{\mathrm{c}}} r^{l-1} \sum_{m=-l}^{l} q_{lm}^{(j)}(r) Y_{lm}(\boldsymbol{\omega})$$

has L_{c} derivatives at $\mathbf{r}=\mathbf{R}_j$, and can be efficiently represented in the form (3.6). Thus, to construct the representation (4.6), we proceed as in Section 4.1 and first construct functions $g_j(\mathbf{r})$, $j=1,\ldots,J$, such that

$$V(\mathbf{r}) f(\mathbf{r}) - g_j(\mathbf{r}) = \|\mathbf{r}-\mathbf{R}_j\|^{-1} \mathcal{O}(\epsilon) \quad \text{for} \quad \|\mathbf{r}-\mathbf{R}_j\| \leq R_{0,j}. \tag{4.7}$$

Once $g_j(\mathbf{r})$ $j=1,\ldots,J$ are computed, we construct the smooth part as

$$\begin{aligned} g_{\mathrm{s}}(\mathbf{r}) &= V(\mathbf{r}) f(\mathbf{r}) - \sum_{j=1}^{J} g_j(\mathbf{r}) \\ &= \sum_{n=1}^{N_{\mathrm{s}}} K_n^{\mathrm{s}}\left(\frac{\mathbf{r}}{\|\mathbf{r}\|}\right) \sum_k a_{k,n} \exp\left(-\alpha_{k,n}\|\mathbf{r}\|\right) + \|\mathbf{r}-\mathbf{R}_j\|^{-1} \mathcal{O}(\epsilon) \end{aligned}$$

Putting all this together, we have

$$V(\mathbf{r}) f(\mathbf{r}) = g_{\mathrm{s}}(\mathbf{r}) + g_{\mathrm{c}}(\mathbf{r}) + \sum_{j=1}^{J} \|\mathbf{r}-\mathbf{R}_j\|^{-1} \mathcal{O}(\epsilon).$$

We now describe the computation of $g_j(\mathbf{r})$ in more detail. For each spherical quadrature node $\boldsymbol{\omega}_n^{\mathrm{c}}$, $n=1,2,\ldots,N_{\mathrm{c}}$, we construct, via Algorithm 1, exponential representations,

$$V(r\boldsymbol{\omega}_n^{\mathrm{c}}+\mathbf{R}_j) f(r\boldsymbol{\omega}_n+\mathbf{R}_j) - g_j(r\boldsymbol{\omega}_n^{\mathrm{c}}+\mathbf{R}_j) = r^{-1}\mathcal{O}(\epsilon), \quad 0 \leq r,$$

where

$$g_j\left(r\boldsymbol{\omega}_n^{\mathrm{c}}+\mathbf{R}_j\right)=\frac{1}{r}\sum_k a_{k,n}^{(j)}\exp\left(-\alpha_{k,n}^{(j)}r\right)+\sum_n b_{k,n}^{(j)}\exp\left(-\beta_{k,n}^{(j)}r\right).$$

Then extending $g_j(\mathbf{r})$ by interpolation,

$$g_j\left(r\boldsymbol{\omega}+\mathbf{R}_j\right)=\sum_{n=1}^{N_{\mathrm{c}}}K_n^{\mathrm{c}}(\boldsymbol{\omega})\left(\sum_k a_{k,n}^{(j)}\exp\left(-\alpha_{k,n}^{(j)}r\right)+\sum_n b_{k,n}^{(j)}\exp\left(-\beta_{k,n}^{(j)}r\right)\right),$$

we arrive at the representation (4.6). The computation of the smooth part $g_{\mathrm{s}}(\mathbf{r})$ proceeds in the same way as the computation of the smooth part of $h_{\mathrm{s}}(\mathbf{r})$ of the multiplicand $f(\mathbf{r})\,g(\mathbf{r})$ in Section 4.1.

4.3. **Convolution with the bound-state Helmholtz and Poisson kernels.** We first discuss how to evaluate the convolution operator,

$$\left(-\Delta+\mu^2\right)^{-1}f(\mathbf{r})=\frac{1}{4\pi}\int_{\mathbb{R}^3}\frac{e^{-\mu\|\mathbf{r}-\mathbf{y}\|}}{\|\mathbf{r}-\mathbf{y}\|}f(\mathbf{y})\,d\mathbf{y}, \tag{4.8}$$

to obtain the representation in the form described in Section 3.1. We assume that $f(\mathbf{r})$ is already given in this form and, for now, $\mu>0$. As is typical in pseudo-spectral methods, it is more efficient to apply this operator in the Fourier domain—in this case, the space of spherical harmonics. Thus, we first convert $f(\mathbf{r})$ from an interpolating representation to a spherical harmonic representation, as described in Section 3.2.

By linearity and translation invariance of $\left(-\Delta+\mu^2\right)^{-1}$, it also suffices to consider $f(\mathbf{r})$ in the form $f(\mathbf{r})=f_0(r)\,Y_l^m(\boldsymbol{\omega})$. Now, by expanding the kernel in (4.8) via spherical harmonics, it can be shown that

$$\left(-\Delta+\mu^2\right)^{-1}\left(f_0(r)\,Y_l^m(\boldsymbol{\omega})\right)=Y_l^m(\boldsymbol{\omega})\,F_l(r),$$

where $F_l(r)$ is given by

$$F_l(r)=\frac{2\mu}{\pi}\left(k_l(r\mu)\int_0^r i_l(\rho\mu)\,f_0(\rho)\,\rho^2d\rho+i_l(r\mu)\int_r^\infty k_l(\rho\mu)\,f_0(\rho)\,\rho^2d\rho\right), \tag{4.9}$$

and $i_l(z)$ and $k_l(z)$ are defined in terms of the modified Bessel functions

$$i_l(z)=\sqrt{\frac{\pi}{2z}}I_{l+1/2}(z),\quad k_l(z)=\sqrt{\frac{\pi}{2z}}K_{l+1/2}(z).$$

Thus, computing a representation of $\left(-\Delta+\mu^2\right)^{-1}f(\mathbf{r})$ simply involves using Algorithm 1 to construct an exponential representation of $F_l(r)$. We also note that when $f(\mathbf{r})$ is of the intermediate form (4.6), the only difference is that we also have integrals of the form

$$F_l(r)=\frac{2\mu}{\pi}\left(k_l(r\mu)\int_0^r i_l(\rho\mu)\,f_0(\rho)\,\rho d\rho+i_l(r\mu)\int_r^\infty k_l(\rho\mu)\,f_0(\rho)\,\rho d\rho\right),$$

for which we also need to construct exponential representations.

Recall that Algorithm 1 requires sampling $F_l(\rho)$ on an equispaced grid $r_n=Rn/(2N)$, $m=0,\ldots,2N-1$; here R is chosen large enough that $|F_l(R)|\leq\epsilon$, where ϵ is the desired approximation error. To do so efficiently, we use the recursion

$$I_1(r_{n+1})\quad\equiv\quad\int_0^{r_{n+1}}i_l(\rho\mu)\,f_0(\rho)\,\rho^2d\rho=I_1(r_n)+\int_{r_n}^{r_{n+1}}i_l(\mu\rho)\,f_0(\rho)\,\rho^2d\rho.$$

The integral between r_n and r_{n+1} can be accurately evaluated using a small number of quadrature points since $i_l(\mu\rho)$ does not oscillate for $\mu > 0$ (e.g., we use 5 quadrature nodes in our examples in Section 5). Once the samples $I_1(r_n)$ are computed, samples of the first integral in (4.9) may be readily obtained. We note that the values of $i_l(\mu\rho)$ at the quadrature points may be computed using an interpolation table constructed off-line. Also, by assumption the function $f_0(\rho)$ is represented with a near optimally small number of exponents, and is thus also inexpensive to sample. The second integral defining $F_l(r)$ may be efficiently sampled in a similar manner.

Computing a representation of $\Delta^{-1}f$ (i.e., in the case $\mu = 0$) proceeds in a similar manner. In particular, we have

$$\Delta^{-1}\left(Y_l^m(\boldsymbol{\omega})\, f_0(r)\right) = Y_l^m(\boldsymbol{\omega})\, F_l(r),$$

where

$$F_l(r) = -\frac{1}{2l+1}\left(r^{-l-1}\int_0^r f_0(r)\,\rho^{2+l}d\rho + r^l\int_r^\infty f_0(r)\,\rho^{-l+1}d\rho\right). \tag{4.10}$$

One technical difference is that $F_l(r)$ decays like $A_l r^{-l-1}$, where A_l depends on the exponents and coefficients of $f_0(r)$; thus, for small l, the direct application of Algorithm 1 would require a prohibitively large number of samples for a representation on the entire half line $r \geq 0$. However, in the quantum chemistry applications considered here, we only need to represent products of the form $g\left(\Delta^{-1}f\right)$, where the function $g(\mathbf{r})$ (and, thus, the overall product) decays exponentially fast. Therefore, it suffices to construct an exponential approximation to $F_l(r)$ only within the numerical support of $g(\mathbf{r})$, which is the approach taken in the examples in Section 5.

Remark 4.1. We can alternatively use the methods in [8] and [20] to construct an efficient exponential representation of $F_l(r)$ on the entire half line $r \geq 0$. In fact, the slowly decaying asymptotic part $A_l r^{-l-1}$ can be represented with a small number of decaying exponentials via the discretization of an appropriate quadrature formula [8]. Once this exponential representation for $A_l r^{-l-1}$ is available, an exponential representation for the rapidly decaying function $F_l(r) - A_l r^{-l-1}$ can be constructed using Algorithm 1.

5. Example of solving Hartree-Fock equations for diatomic molecules

We now use the representations and algorithms in Sections 3.1 and 4 to solve the Hartree-Fock equations for several diatomic molecules.

Example 1. As our first example of using the representations and algorithms discussed above, we solve the Hartree-Fock equation,

$$\left(-\frac{1}{2}\Delta + V - 4\pi\Delta^{-1}\left(|\phi|^2\right)\right)\phi = E\phi, \tag{5.1}$$

with the potential

$$V(\mathbf{r}) = \frac{Z_1}{\|\mathbf{r}-\mathbf{R}_1\|} + \frac{Z_2}{\|\mathbf{r}-\mathbf{R}_2\|}.$$

As in [19, 31, 32], our basic approach involves recasting (5.1) as an integral equation which we solve via iteration. However, in contrast to [19, 31, 32], we represent $\phi(\mathbf{r})$ as described in Section 3.1 and apply algorithms described in Section 4. Since the

spatial orbital $\phi(\mathbf{r})$ has cusp-like singularities at the nuclei $\mathbf{r} = \mathbf{R}_1$ and $\mathbf{r} = \mathbf{R}_2$, the Hartree-Fock equation provides a useful accuracy test for our approach.

Let us now describe the solution method in greater detail. We write (5.1) as

$$\left(-\Delta + \mu^2\right)\phi = -2V_\phi\phi,$$

where $\mu^2 = -2E$ and

$$V_\phi = V - 4\pi\Delta^{-1}\left(|\phi|^2\right).$$

We solve this via the following iteration:

$$\begin{aligned}
\widetilde{\phi} &\leftarrow -2\left(-\Delta + \mu^2\right)^{-1}(V_\phi\phi), \\
E &\leftarrow E + \frac{\left\langle \phi - \widetilde{\phi}, V_\phi\phi \right\rangle}{\left\|\widetilde{\phi}\right\|^2}, \\
\phi &\leftarrow \frac{\widetilde{\phi}}{\left\|\widetilde{\phi}\right\|}, \\
\mu &\leftarrow \sqrt{-2E}.
\end{aligned}$$

The inner products

$$\langle f, g\rangle = \int_{\mathbb{R}^3} f(\mathbf{r})\, g(\mathbf{r})\, d\mathbf{r},$$

are computed by representing $f(\mathbf{r})\, g(\mathbf{r})$ in the basic form (3.5) and (3.6) using the algorithm in Section 4.1, and computing the resulting integrals analytically (a similar comment applies to computing $\langle f, Vg\rangle$). We let the above iteration run until the computed correction to the orbital energy E is less than the desired accuracy.

We solve equation (5.1) with $Z_1 = -1$, $Z_2 = -2$ and $\mathbf{R}_1 = (0, 0, -.7)$ and $\mathbf{R}_2 = (0, 0, .7)$, corresponding to the Helium Hydride Ion HeH+. We use $N_{\mathrm{c}} = 12$ spherical quadrature nodes for the cusp part, $N_{\mathrm{s}} = 192$ nodes for the smooth part and a tolerance of $\epsilon = 10^{-7}$ in Algorithm 1. We note that the number of radial samples needed when using Algorithm 1 is typically a factor of ≈ 10 more than the number of resulting exponential/coefficient pairs (and varies depending on the spherical quadrature direction).

We verify that, for this choice of parameters, the orbital energy $E = -1.660544$ is computed with six significant digits, and with 4.4×10^{-7} absolute error. The comparison is made using the MADNESS software [12] yielding the orbital energy $E = -1.66054378$. The total number of exponent/coefficient pairs for the representation of the orbital $\phi(\mathbf{r})$ (in the form (3.6)) is 637.

Remark 5.1. We have implemented Example 1 in Fortran 90 and made a preliminary speed comparison with the MADNESS code. MADNESS uses a similar iteration scheme, but is based on an adaptive basis representation via multiwavelets [19]. In our comparison, we choose parameters in both codes to achieve relative error $\approx 10^{-8}$ for the orbital energy E. Comparing timings for this example, the approach in this paper is about 2.5 times faster than the MADNESS code. We made no attempt to optimize our implementation, in part because the algorithms presented here are likely to be improved upon as we expect some modifications (outlined in Section 6) to speed up our initial implementation by an additional factor. We plan to do a careful speed comparison in a separate publication.

Example 2. For our second example, we consider the Hartree-Fock equation for Lithium Hydride, LiH. We have

$$\mathcal{F}\phi_j(\mathbf{r}) = E_j\phi_j(\mathbf{r}), \quad j = 1, 2, \tag{5.2}$$

where $\mathcal{F} = -\frac{1}{2}\Delta + V + 2J - K$,

$$J\phi_j = \phi_j\left(-4\pi\Delta^{-1}\left(|\phi_1|^2 + |\phi_2|^2\right)\right),$$

$$K\phi_j = \phi_1\left(-4\pi\Delta^{-1}\left(\phi_1^*\phi_j\right)\right) + \phi_2\left(-4\pi\Delta^{-1}\left(\phi_2^*\phi_j\right)\right),$$

and

$$V(\mathbf{r}) = \frac{Z_1}{\|\mathbf{r} - \mathbf{R}_1\|} + \frac{Z_2}{\|\mathbf{r} - \mathbf{R}_2\|}.$$

The iteration is the same as in the first example, with two modifications. First, the spatial orbitals $\phi_1(\mathbf{r})$ and $\phi_2(\mathbf{r})$ are orthogonalized after each iteration. Second, the orbital energies E_1 and E_2 are updated after each iteration by solving for the eigenvalues of the 2×2 matrix $H_{ij} = \langle \mathcal{F}\phi_i, \phi_j \rangle$. In order to compute the inner products of the form $\int \psi\Delta\phi\mathbf{dr}$, we use a straightforward modification of the algorithms in Sections 4.1 and 4.2 to represent of $\psi\Delta\phi$ in the basic functional form of Section 3.1. As in Section 4.2, the $1/r^2$-type singularities at $\mathbf{r} = \mathbf{R}_1$ and $\mathbf{r} = \mathbf{R}_2$ are carried explicitly. Once the representation for $\psi\Delta\phi$ is constructed, the integral is evaluated analytically.

In this example, we again use $N_c = 12$ spherical quadrature nodes for the cusp part and $N_s = 192$ nodes for the smooth part. We also take $\mathbf{R}_1 = (-3.15/2, 0, 0)$ and $\mathbf{R}_2 = (3.15/2, 0, 0)$, and again choose a tolerance of $\epsilon = 10^{-7}$ in Algorithm 1.

The computed orbital energies $E_1 = 2.4517624$ and $E_2 = 0.2978234$ agree to six significant digits with the values $E_1 = 2.451763$ and $E_2 = 0.297823$ computed by the MADNESS software [12], and have absolute errors of 5.2×10^{-7} and 4.1×10^{-7}. The computed total energy $E_{\text{tot}} = -7.986937$ also agrees to six significant digits with the value $E_{\text{tot}} = 7.9869364$ computed in MADNESS, and has an absolute error of 7.7×10^{-7}. The total number of exponent/coefficient pairs for the orbitals $\phi_1(\mathbf{r})$ and $\phi_2(\mathbf{r})$ (in the form (3.6)) is $1,282$ and $1,327$.

Figure 5.1 displays the spatial orbital $\psi(\mathbf{r}) \equiv \phi_2(\mathbf{r})$, the cusp part $\psi_c(\mathbf{r})$, and the smooth part $\psi_s(\mathbf{r})$, on the line $\mathbf{r} = (x, 0, 0)$ connecting the two nuclei locations $\mathbf{R}_1$ and $\mathbf{R}_2$ (plots for the other spatial orbitals for LiH and for HeH+ are similar).

6. Alternative formulations

We are currently considering several possible modifications of our approach.

(1) It may be possible to modify our approach so that, instead of using Algorithm 1, we apply the reduction algorithm in [20]. This change will likely result in a faster algorithm, and is currently being explored.
(2) It may be desirable to use only the cusp part (i.e. (3.2) and (3.5)) in the functional representations; this can be achieved with minor modifications to the algorithms in Section 4 by using a partition of unity (see also e.g. [29]),

$$\sum_{j=1}^{J} \chi_j(\mathbf{r}) = 1,$$

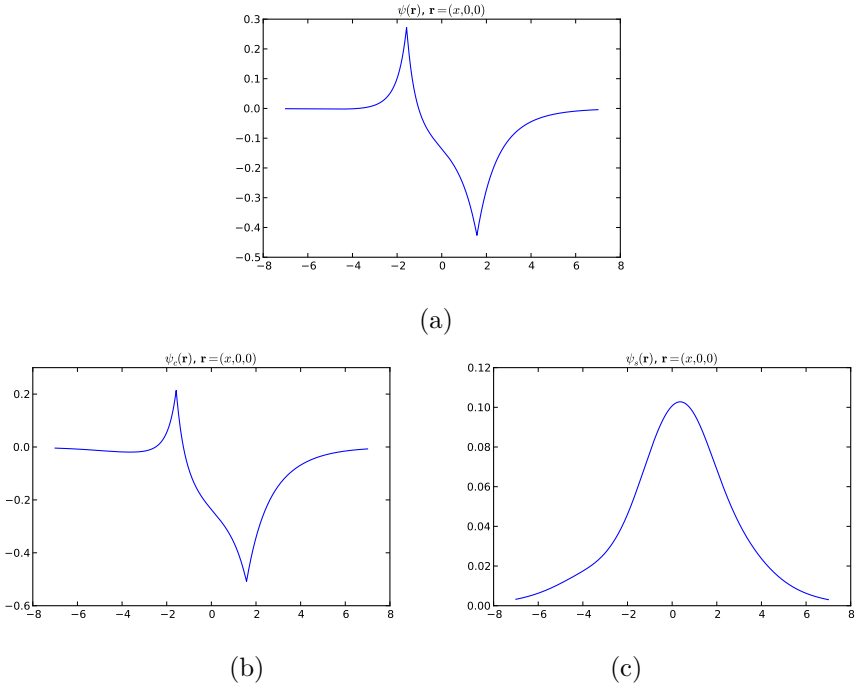


FIGURE 5.1. The spatial orbital $\psi(\mathbf{r}) \equiv \phi_2(\mathbf{r})$ for LiH on the line $\mathbf{r} = (x, 0, 0)$ connecting $\mathbf{R}_1$ and $\mathbf{R}_2$ (a), and the corresponding cusp part $\psi_c(\mathbf{r})$ (b) and the smooth part $\psi_s(\mathbf{r})$ (c).

where the function $\chi_j(\mathbf{r})$ vanishes at the singularity locations $\mathbf{R}_k \neq \mathbf{R}_j$. This approach may also result in more efficient representations and algorithms.

(3) It may be possible to speed up the algorithms by performing the nonlinear approximations on a sparse subset of the spherical quadrature nodes $\boldsymbol{\omega}_n^s$ (and treat the resulting collection of exponentials as a fixed basis for the remaining spherical quadrature nodes). Effectively, this implies a multiresolution approach for computing on the sphere, a development that has many additional applications.

(4) Instead of using decaying exponentials in (1.1) and (1.2), we can use decaying Gaussians with complex-valued exponents. Although such a switch will increase the number of terms in the orbital representations, working with Gaussians should simplify the computations in a number of ways. We plan to examine this trade-off.

Acknowledgement. We thank Dr. Robert Harrison for sharing with us his insight into computational quantum chemistry, for his many helpful suggestions and comments, and for his assistance in using the MADNESS software. We also thank Dr. Martin Mohlenkamp for his careful reading of the manuscript, and for the help in comparing our method with that based on multiwavelets.

References

[1] V. M. Adamjan, D. Z. Arov, and M. G. Kreĭn. Infinite Hankel matrices and generalized Carathéodory-Fejér and I. Schur problems. *Funkcional. Anal. i Priložen.*, 2(4):1–17, 1968.

[2] V. M. Adamjan, D. Z. Arov, and M. G. Kreĭn. Infinite Hankel matrices and generalized problems of Carathéodory-Fejér and F. Riesz. *Funkcional. Anal. i Priložen.*, 2(1):1–19, 1968.

[3] V. M. Adamjan, D. Z. Arov, and M. G. Kreĭn. Analytic properties of the Schmidt pairs of a Hankel operator and the generalized Schur-Takagi problem. *Math. USSR Sbornik*, 15(1):34–75, 1971.

[4] C. Ahrens and G. Beylkin. Rotationally Invariant Quadratures for the Sphere. *Proceedings of the Royal Society A*, 465:3103–3125, 2009.

[5] G. Beylkin, M. J. Mohlenkamp, and F. Pérez. Preliminary results on approximating a wavefunction as an unconstrained sum of slater determinants. *Proceedings in Applied Mathematics and Mechanics*, 7(1):1010301–1010302, 2007. Special Issue: Sixth International Congress on Industrial Applied Mathematics (ICIAM07) and GAMM Annual Meeting, Zürich 2007.

[6] G. Beylkin, M. J. Mohlenkamp, and F. Pérez. Approximating a wavefunction as an unconstrained sum of Slater determinants. *Journal of Mathematical Physics*, 49(3):032107, 2008.

[7] G. Beylkin and L. Monzón. On approximation of functions by exponential sums. *Appl. Comput. Harmon. Anal.*, 19(1):17–48, 2005.

[8] G. Beylkin and L. Monzón. Approximation of functions by exponential sums revisited. *Appl. Comput. Harmon. Anal.*, 2009. In press.

[9] S. F. Boys. The integral formulae for the variational solution of the molecular many-electron wave equations in terms of gaussian functions with direct electronic correlation. *Proceedings of the Royal Society of London Series a- Mathematical and Physical Sciences*, 258(1294):402, 1960. Proc. R. Soc. London Ser. A-Math.

[10] P. E. Cade and W. M. Huo. Electronic structure of diatomic molecules. VII.A. Hartree—Fock wavefunctions and energy quantities for the ground states of the second-row hydrides, AH. *The Journal of Chemical Physics*, 47(2):649–672, 1967.

[11] H. Cheng, Z. Gimbutas, P.-G. Martinsson, and V. Rokhlin. On the compression of low-rank matrices. *SIAM Journal of Scientific Computing*, 205(1):1389–1404, 2005.

[12] G. Fann, R. Harrison, G. Beylkin, R. Hartman-Baker, J. Jia, W. A. Shelton, and S. Suziki. MADNESS in chemistry and physics. *J. of Physics, Conference Series*, 2007.

[13] J. Fosso-Tande and R.J. Harrison. Implicit solvation models in a multiresolution multiwavelet basis. *Chemical Physics Letters*, to appear, 2013.

[14] I.S. Gradshteyn and I. M. Ryzhik. *Table of Integrals, Series, and Products*. Academic Press, 5th edition, 1994.

[15] N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Review*, 53(2):217–288, 2011.

[16] R. Harrison, G. Fann, Z. Gan, T. Yanai, S. Sugiki, A. Beste, and G. Beylkin. Multiresolution computational chemistry. *J. of Physics, Conference Series*, 2005.

[17] R.J. Harrison, G.I. Fann, T. Yanai, and G. Beylkin. Multiresolution quantum chemistry in multiwavelet bases. In P.M.A. Sloot et. al., editor, *Lecture Notes in Computer Science. Computational Science-ICCS 2003*, volume 2660, pages 103–110. Springer, 2003.

[18] R.J. Harrison, G.I. Fann, T. Yanai, Z. Gan, and G. Beylkin. Multiresolution quantum chemistry: basic theory and initial applications. APPM preprint 516, Univ. of Colorado, December 2003.

[19] R.J. Harrison, G.I. Fann, T. Yanai, Z. Gan, and G. Beylkin. Multiresolution quantum chemistry: basic theory and initial applications. *J. Chem. Phys.*, 121(23):11587–11598, 2004.

[20] T. S. Haut and G. Beylkin. Fast and accurate con-eigenvalue algorithm for optimal rational approximations. *SIAM J. Matrix Anal. Appl.*, 33(4):1101–1125, 2012. http://dx.doi.org/10.1137/110821901, see also arXiv:1012.3196 [math.NA].

[21] T. S. Haut, G. Beylkin, and L. Monzón. Solving Burgers' equation using optimal rational approximations. *Appl. Comput. Harmon. Anal.*, 34:83–95, 2013. http://dx.doi.org/10.1016/j.acha.2012.03.004 (electronic).

[22] V. I. Lebedev. Quadratures on a sphere. *Zh. vychisl. Mat. mat. Fiz.*, 16(2):293–306, 1976.

[23] V.I. Lebedev and D.N. Laikov. A quadrature formula for the sphere of the 131st algebraic order of accuracy. *Doklady Mathematics*, 59(3):477–481, 1999. Translated from Doklady Akademii Nauk, Vol. 36, No. 6, 1999, pp. 741-745.

[24] E. Liberty, F. Woolfe, P-G. Martinsson, V. Rokhlin, and M. Tygert. Randomized algorithms for the low-rank approximation of matrices. *Proc. Natl. Acad. Sci. USA*, 104(51):20167–20172, 2007.

[25] J.V.L. Longstaff and K. Singer. The use of gaussian (exponential quadratic) wave functions in molecular problems. ii. wave functions for the ground state of the hydrogen atom and of hydrogen molecule. *Proceedings of the Royal Society of London Series A- Mathematical and Physical Sciences*, 258(1294):421, 1960. Proc. R. Soc. London Ser. A-Math.

[26] J. N. Murrell and C. L. Silk. Optimum atomic orbital exponents for molecular wave functions. *Symp. Faraday Soc.*, 2:84–94, 1968.

[27] F. A. Pahl and N. C. Handy. Plane waves and radial polynomials: a new mixed basis. *Molecular Physics*, 100:3199–3224, 2002.

[28] M. Reynolds, G. Beylkin, and L. Monzón. On generalized Gaussian quadratures for bandlimited exponentials. *Appl. Comput. Harmon. Anal.*, 2013. In press, http://dx.doi.org/10.1016/j.acha.2012.07.002.

[29] T. Shiozaki and S. Hirata. Grid-based numerical hartree-fock solutions of polyatomic molecules. *Phys. Rev. A*, 76:040503, Oct 2007.

[30] K. Singer. The use of gaussian (exponential quadratic) wave functions in molecular problems. i. general formulae for the evaluation of integrals. *Proceedings of the Royal Society of London Series a- Mathematical and Physical Sciences*, 258(1294):412, 1960. Proc. R. Soc. London Ser. A-Math.

[31] T. Yanai, G.I. Fann, Z. Gan, R.J. Harrison, and G. Beylkin. Multiresolution quantum chemistry: Analytic derivatives for Hartree-Fock and density functional theory. *J. Chem. Phys.*, 121(7):2866–2876, 2004.

[32] T. Yanai, G.I. Fann, Z. Gan, R.J. Harrison, and G. Beylkin. Multiresolution quantum chemistry: Hartree-Fock exchange. *J. Chem. Phys.*, 121(14):6680–6688, 2004.

[33] T. Yanai, R.J. Harrison, and N.C. Handy. Multiresolution quantum chemistry in multiwavelet bases: time-dependent density functional theory with asymptotically corrected potentials in local density and generalized gradient approximations. *Mol. Phys*, 103, 2005.

Department of Applied Mathematics, University of Colorado, Boulder, CO 80309-0526, United States, *T.S.Haut current affiliation: Los Alamos National Lab, NM 87545, United States