Linear scaling density fitting

Alex Sodt, Joseph E. Subotnik, and Martin Head-Gordon[a)]
Department of Chemistry, University of California, Berkeley, California 94720
and Chemical Sciences Division, Lawrence Berkeley National Laboratory, Berkeley, California 94720

(Received 7 August 2006; accepted 28 September 2006; published online 17 November 2006)

Two modifications of the resolution of the identity (RI)/density fitting (DF) approximations are presented. First, we apply linear scaling and J-engine techniques to speed up traditional DF. Second, we develop an algorithm that produces local, accurate fits with effort that scales linearly with system size. The fits produced are continuous, differentiable, well-defined, and do not require preset fitting domains. This metric-independent technique for producing *a priori* local fits is shown to be accurate and robust even for large systems. Timings are presented for linear scaling RI/DF calculations on large one-, two-, and three-dimensional carbon systems. © *2006 American Institute of Physics*. [DOI: 10.1063/1.2370949]

I. INTRODUCTION

In this paper we propose and implement a method for the calculation of the Coulomb interactions for self-consistent field (SCF) calculations, such as Hartree-Fock[1] (HF) theory or Kohn-Sham (KS) density functional theory[2] (DFT). The effort of Coulomb matrix construction is most significant for DFT, as it lacks the more expensive and nonlocal HF exchange, though these steps have been made linear scaling for insulators.[3,4] DFT instead traditionally employs a numerical quadrature scheme for exchange-correlation evaluation, for which linear scaling has been achieved.[5] DFT is widely used in the computational chemistry community due to its arguably unrivaled combination of efficiency and accuracy. As a computationally significant step of DFT, the Coulomb problem for SCF calculations has received much attention recently, with pivotal breakthroughs allowing for linear scaling with system volume using a tree-based multipole method[6–8] and efficiency with respect to basis set size using the pseudospectral algorithm[9] or fitting methods,[10] for example. An alternative to what we consider in this paper, the pseudospectral method developed by Murphy *et al.* evaluates interactions in real space on a grid. Also, the Gaussian and augmented plane wave[11] (GAPW) method of Lippert *et al.* uses Fourier techniques to efficiently represent Coulomb interactions. Similar to the pseudospectral and fitting approaches, it achieves great speedups with some sacrifice of accuracy. Related to the GAPW method is the Fourier transform Coulomb[12,13] (FTC) method, which complements multipole methods in that it is able to deal swiftly with diffuse basis functions and without compromising accuracy. As this work is a modification of the long-standing resolution of the identity (RI) or density fitting (DF) scheme, we will devote more text to the history and details of this procedure, although in this paper we attempt to make full use of as many of the Coulomb community's tools as possible.

With DF, one replaces the density with a much simpler, approximate representation. In a typical DFT calculation using Gaussian basis functions, the density is a sum of single particle basis function products. As the single particle basis tends to contain diffuse functions, there are many of these products. With DF, the complicated density is replaced with single coordinate, atom-centered basis functions, termed the auxiliary basis set. Calculation of the Coulomb matrix then proceeds by evaluating the interaction of the function products with the fit density, a procedure that requires far fewer fundamental electron repulsion integrals, since the number of auxiliary functions is far less than the number of function products.

We will present two efficient DF algorithms. The first uses the full fitting procedure (costing N^3 CPU and N^2 storage) but has linear scaling interactions (typically the bottleneck step). The second includes a modification to the standard fitting procedure that seeks to retain the accuracy of DF while making the entire fitting and interaction process linear scaling, using fitting function subsets that vary continuously with atomic motion. Before we explain the details, however, we wish to give more background on DF.

II. DENSITY FITTING

Fitting techniques for two-electron integrals date back to early attempts to compute molecular properties.[14–16] In most approaches, a measure of the error is chosen and minimized. Two primary *fitting metrics* have been tested and applied over the past few decades: the overlap metric and the Coulomb metric.[17–19] The fitting metric determines in what sense the fit is good. The overlap metric determines the fit density that has least-squares minimal deviations locally, while the Coulomb metric minimizes least-squares deviations of Coulomb interactions.[20] With both metrics, one can perform a least-squares minimization[21,22] of the density minus its fit, overlapping (or in the case of the Coulomb metric, one could say interacting) with itself,

[a)]Electronic mail: mhg@bastille.cchem.berkeley.edu

$$\int (\rho(r_1) - \tilde{\rho}(r_1)) m(r_1, r_2) (\rho(r_2) - \tilde{\rho}(r_2)). \quad (1)$$

Here in the case of the overlap metric, $m(r_1, r_2)$ is $\delta(r_1 - r_2)$, and in the case of the Coulomb metric it is $1/|r_1 - r_2|$. This leads to the following expression for the fit coefficients, C_ρ^K:

$$C_\rho^K = (K|L)_m^{-1} (L|\rho)_m, \quad (2)$$

where the subscript m indicates that the integral was performed in that fitting metric. In this paper we will use $(\alpha|\beta)$ to indicate the Coulomb interaction of function α and function β:

$$(\alpha|\beta) = \int \int \alpha(\mathbf{r}_1) |\mathbf{r}_1 - \mathbf{r}_2|^{-1} \beta(\mathbf{r}_2) d\mathbf{r}_1 d\mathbf{r}_2. \quad (3)$$

To solve the least-squares fit equation, one inverts the fitting basis interaction matrix. Efficient Cholesky decomposition[23] of the fitting basis interaction matrix requires work that scales to the third power of system size. Third order work of this sort is common in quantum chemical calculations, for example, in the diagonalization of the Fock matrix and in forming an orthonormal basis from a set of atomic orbitals. However, for reasonable accuracy, the number of fitting functions should be approximately three to four times larger than the number of standard basis functions. An inversion of a fitting basis matrix could thus be 100 times more expensive than a similar operation on an atomic orbital matrix and require much more memory. With a linear scaling density update step,[24] for large systems, performing the fitting procedure could be the bottleneck step. Sierka *et al.* have performed efficient RI calculations for Coulomb builds using multipole techniques,[10] but their algorithm is not linear scaling, due primarily to the fitting step. Our primary goal in the second part of this work is to develop a linear scaling procedure for density fitting, though the benefits of fit locality could have an impact on correlated calculations and Hartree-Fock exchange with the RI approximation.

Given the fit to the density, C_ρ^K, one may compute the interactions of a function product ($\mu\nu$, for example) with the density

$$J_{\mu\nu} = \sum_K (\mu\nu|K) C_\rho^K. \quad (4)$$

Via the SCF algorithm, a new density would be calculated from the resulting Fock matrix, and the fitting procedure would repeat. Notice that three center $(\mu\nu|K)$ integrals were required twice per iteration; once for the fit [Eq. (2)] and again for interactions [Eq. (4)]. However, both of these steps are J-type steps, insomuch as they are some function, $\mu\nu(r)$ or $K(r)$, interacting with some density, $\rho(r)$ or $\tilde{\rho}(r)$. The J-engine[25–27] technique may be applied in these cases. We refer the interested reader to the cited works for the intricate details. In summary, instead of computing $(\mu\nu|K)$ explicitly, $\mu\nu(r)$ and $K(r)$ are expanded in Hermite polynomials p and q, respectively. J_p is constructed (in effect but not actually in practice) from the contraction of $(p|q)$ with P_q, which is then postprocessed to the desired $J_{\mu\nu}$.

With the J-engine, fundamental Coulomb integrals must be computed in both the fitting and interaction passes of DF, but the more expensive density dependent recurrence steps are different for each pass. This is in contrast to the algorithm without J-engine, in which $(\mu\nu|K)$ integrals are computed twice each iteration. The J-engine would be applied to computing J for the bra in the fitting pass and for the ket in the interaction pass. Ignoring the cost of the fundamental Coulomb integrals, the cost of J-engine building would be the same if the fitting and interaction passes took place simultaneously or not.

TABLE I. CPU time for building a single J matrix, as well as performing the auxiliary basis inversion required for fitting. The basis set used is TZ (Ref. 28), along with the corresponding universal fitting basis (Ref. 29).

Alkane chain length	J build	Inversion
10	2.3	0.2
20	5.5	1.2
40	11.8	9.7
120	43.1	177.3
200	77.8	738.6

Linear scaling of the three-center integral computation is accomplished using the continuous fast multipole method[7] (CFMM). Sierka *et al.* have used a similar approach using atom-center multipole expansions to accelerate so-called far-field interactions to great success. In this section we will present timings of Coulomb matrix construction with and without the J-engine to highlight its efficacy. The system we consider will be linear alkanes, a case for which the fitting procedure has a greater relative cost. In a summary of our linear scaling DF work, later on we will present timings for two- and three-dimensional systems.

In this paper, all calculations were run on one processor of a 2.2 GHz Xserve G5, using up to 8 Gbytes of RAM. Convergence thresholds for the SCF procedure were set to 10^{-6}, and thresholds for function product significance were set to 10^{-8}. In all cases we screened the change of elements of the density matrix to avoid integral computation, making later SCF iterations cheaper than earlier ones. The quoted times will be averages over the SCF computation.

Table I shows the timings of our first DF algorithm for linear alkanes, including the time required to perform the inverse of the auxiliary interaction matrix. We also tested the J-build speedup due to using the J-engine for a DF calculation and for a standard CFMM calculation. These are shown in Table II. The basis set for both calculations is TZ,[28] a triple zeta basis set.

TABLE II. Speedup factor for using the J engine in both a DF calculation and an exact calculation of the J matrix. The basis set used is TZ (Ref. 28), along with the corresponding universal fitting basis (Ref. 29).

Alkane chain length	DF speedup	Exact speedup
10	1.53	1.49
20	1.56	1.34
40	1.60	1.24
120	1.48	1.21
200	1.49	1.17

III. LOCAL ATOMIC DENSITY FITTING (LADF)

The long-ranged properties of the Coulomb metric make density fitting for large systems troublesome. The effect of the Coulomb operator dies off very slowly as two charge distributions become separated. Indeed, it is believed that the Coulomb metric allows the electric field generated by a charge distribution to be fitted rather than the actual density itself. In terms of absolute accuracy, the overlap operator has been shown to be inferior to the Coulomb operator,[30] though the study did not use variational fitting as suggested by Dunlap.[31] Jung *et al.* recently proposed[32] an alternative metric for fitting Coulomb integrals, the attenuated Coulomb metric, $\mathrm{erfc}(\omega r)r^{-1}$, where the tuning parameter ω can vary between zero (the Coulomb metric) and ∞ (effectively the overlap metric).

As Jung *et al.* show, a local fitting metric may determine a more local fit. It may be possible to perform a linear scaling inverse in an attenuated metric, for example, using the AINV procedure of Benzi, Meyer, and Tůma,[33] though the accuracy of such a procedure may not be consistent. In the absence of a linear scaling sparse inverse, a full-scale inversion must still be performed to determine the local fit coefficients in an attenuated metric. As an alternative approach to generating local fits, we employ a "bump function" to make irrelevant those far-away functions that we do not want involved in the fit. The bump function we use is a continuous, differentiable function, which varies from zero to one across some finite distance. By fitting the density piecewise, functions far away from the density may be ignored and not computed in the inverse, so will play no role in the fit. As the nuclei rearrange (for example, in a geometry optimization or for molecular dynamics), the far-away functions are gradually brought into relevance by the bump function in such a way that gradients and forces are always meaningful (if perhaps not as accurate as for the full density fitting procedure).

One important technique that will be used in this work was advocated by Dunlap. He suggests that for best relative accuracy, fitting should be performed variationally.[31] That is, if the quality of the fit (for example, the error in the interaction energy) is not changed to first order in fit error (the difference between the fit quantity and the target quantity), beneficial cancellation will occur when relative energies are computed. To accomplish this, his directive is that one should correct this minimization functional for first order error in the fit. For example, consider the Coulomb interaction between the charge density ρ and some other density, f,

$$\langle\widetilde{\rho}| = \langle\rho| + \langle\delta_\rho|, \tag{5}$$

$$\langle\widetilde{f}| = \langle f| + \langle\delta_f|. \tag{6}$$

Here the quantities of interest are fits with error δ. Subtracting the interaction of the fitted quanties from the sum of the interactions with only one term fitted yields

$$\langle f|r_{12}^{-1}|\widetilde{\rho}\rangle + \langle\widetilde{f}|r_{12}^{-1}|\rho\rangle - \langle\widetilde{f}|r_{12}^{-1}|\widetilde{\rho}\rangle \tag{7}$$

$$= \langle f|r_{12}^{-1}(|\rho\rangle + |\delta_\rho\rangle) + (\langle f| + \langle\delta_f|)r_{12}^{-1}|\rho\rangle - (\langle f| + \langle\delta_f|)r_{12}^{-1}(|\rho\rangle + |\delta_\rho\rangle) \tag{8}$$

$$= \langle f|r_{12}^{-1}|\rho\rangle - \langle\delta_f|r_{12}^{-1}|\delta_\rho\rangle. \tag{9}$$

This prescription is free of error to first order in the fit. For standard DF using the complete inverse and the Coulomb metric, fits are automatically free of first order error. We call this modification of the fitting procedure the Dunlap correction.

Suppose that, in lieu of the exact inverse, we introduce an approximation that differs from the exact one by δ_{KL}:

$$\widetilde{\langle K|L\rangle}^{-1} = \langle K|L\rangle^{-1} + \delta_{KL}, \tag{10}$$

the previous analysis will be modified as such:

$$\langle\widetilde{\rho}| = \langle\rho| + \langle\delta_\rho| - \delta_{KL}\langle L|\rho\rangle, \tag{11}$$

$$\langle\widetilde{\rho}| = \langle\rho| + \langle\delta'_\rho|. \tag{12}$$

That is, the fitting scheme using an approximate inverse is free of second-order error if the Dunlap correction is used. Depending on how valid the approximate inverse is, the error may be unacceptable, however.

Our scheme for approximating the inverse is motivated by the rationale that nearby fitting functions will be used to fit a localized piece of the density and that the fit will not be altered appreciably by ignoring far-away fitting functions. It closely resembles the method by Fonseca Guerra *et al.*[34] in which atomic pair densities are fitted using auxiliary functions sourced on the atomic pair. Their criticism of traditional DF is that an atom A may be accidentally near another atom B, and help with the fit of atom B's density. This would amount to essentially auxiliary basis set superposition error. However, we believe that much of the robustness of DF is obtained from fitting the Coulomb potential rather than the density. The auxiliary functions on atom A are helpful for fitting the local potential generated by atom B. Atom A will also help improve the interaction of atoms B and C, but this effect will not be as potent. With this in mind, we retain the use of the auxiliary functions of nonparticipatory atoms in the fit of a particular density.

The density is divided into atomic parts that will be fitted according to the above logic. It is necessary that the results of this division be continuous as the atoms move to assure continuity of the potential energy surface. The density itself is a sum of atomic orbital products, multiplied by an atomic orbital density matrix,

$$\rho(r) = \sum_{\mu_A,\nu_B} \mu_A(r - r_A)\nu_B(r - r_B)P_{\mu\nu}, \tag{13}$$

where μ_A is centered on atom A, and ν_B is centered on atom B. The product $\mu_A\nu_B$ is centered on a line connecting r_A and r_B, as a consequence of the Gaussian product rule. We associate with atom A those $\mu_A\nu_B$ products closer to atom A than atom B (and if the product is directly between, multiplied by $\frac{1}{2}$). In the case of contracted basis functions, the primitive products are divided such that they fulfill one criterion. This will define a division of the density that is continuous with respect to nuclear motion.

When computing the fit for a local piece of the density (linked to some target atom), fitting functions are weighted by a bump function that is zero if the source atom of a fitting

function is outside a certain radius of the target atom, r_1. The bump is only applied to blocks of the interaction matrix between atoms, i.e., interactions within an atomic block are not bumped. Finally, the inverted matrix is bumped again, without the off-diagonal restriction.

In this work, we define the bump function b as

$$b(x) = 1, \quad x \leqslant = r_0,$$

$$b(x) = \frac{1}{1 + \exp\{(r_1 - r_0)/(r_1 - x) - (r_1 - r_0)/(x - r_0)\}}, \tag{14}$$

$$x \in (r_0, r_1),$$

$$b(x) = 0, \quad x \geqslant = r_1. \tag{15}$$

We define a bump matrix for some atom X as

$$B_X = \begin{pmatrix} \hat{b}(|R_A - R_X|) & 0 & 0 & \\ 0 & \hat{b}(|R_B - R_X|) & 0 & \cdots \\ 0 & 0 & \hat{b}(|R_C - R_X|) & \\ & \vdots & & \ddots \end{pmatrix}, \tag{16}$$

where $\hat{b}$ refers to an identity matrix in an atomic block multiplied by the scalar bump function and R refers to an atomic position, with R_X referring to the position of the target atom.

The expression for an inverse of an atomic block is then

$$(K|L)_X^{-1} = B_X((K|L)_D + B_X(K|L)_{OD}B_X)^{-1}B_X, \tag{17}$$

where $\langle K|L\rangle_D$ refers to the block diagonal portion of the auxiliary interaction matrix, and $(K|L)_{OD}$ refers to the block off-diagonal portion.

Equation (17) is a natural choice for the local inversion procedure because it fulfills three primary goals. First, it produces a fit that is continuous and differentiable. Each matrix is a smoothly varying function of the atomic coordinates. The inverse is always well defined because the interaction matrix is symmetric positive definite, and the diagonal piece is not bumped until after inversion. Second, it decouples those fitting functions not overlapping considerably with the target density without producing an on-diagonal near-zero value that would befoul the inversion procedure, such as would be the case if the rows and columns of an atom's fitting functions were bumped. Third, it removes completely those fitting functions far enough that we do not wish them impacting the fit at all; this is the responsibility of the outer bump matrices in Eq. (17). If we were not to include the outer bump, the uncoupled fits would add together as if they existed in a vacuum, which they do not. Essentially, the inner bump decouples the blocks to allow linear scaling, and the outer bump removes far-away functions from contributing to the fit. The off-diagonal bumping operation does not explicitly shrink the space of the inversion, but if an atom is outside r_1, it will be completely decoupled from the other blocks and, moreover, completely annihilated by the bump after the inverse. In practice, fitting functions whose source atom is further than r_1 from the target atom are neglected, rendering the inversion procedure linear scaling.

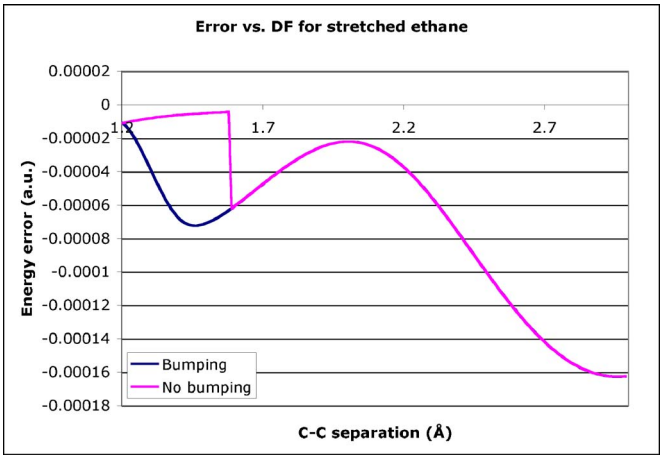

FIG. 1. The error of approximating the inverse in a DF calculation, both with the bump function and without. There is a discontinuity in the unbumped approximate inverse when the carbons in the ethane molecule are separated by r_1 (3 a.u.). The basis set is SV (Ref. 28) and the universal RI-J fitting basis (Ref. 29).

To summarize the fitting procedure, only those fitting functions within r_1 of an atom's density are used to compute the fit. Fitting functions farther away than r_0, but closer than r_1, are modified by a bump function so that they will contribute less as they approach r_1 and more as they approach r_0. Functions farther than r_1 need not be computed because any contribution would be annihilated by the outer bump matrix in Eq. (17). The procedure is linear scaling because a fixed-size inverse is computed for each atom.

The local atomic inverse, $(K|L)_X^{-1}$, is applied in the Dunlap correction, Eq. (7), to evaluate the contribution of an atomic density to the Coulomb interaction within the auxiliary basis

$$(f_B|r_{12}^{-1}|\rho_A) \approx (f_B|K)(K|L)_B^{-1}(L|\rho_A) + (f_B|K)(K|L)_A^{-1}(L|\rho_A) - (f_B|K)(K|M)_B^{-1}(M|N)(N|L)_A^{-1}(L|\rho_A). \tag{18}$$

We suggest using small fitting regions (r_0=4.0 and r_1 =5.0, in a.u.) for their combination of efficiency and robustness. All example calculations will use these values of r_0 and r_1. We will refer to the fitting scheme for building the Coulomb matrix as local atomic density fitting (LADF) and as atomic resolution of the identity (ARI) in general.

As a simple but illustrative test of the error associated with approximating the fit inverse, we took the ethane molecule, with carbons a distance R apart. We then vary R through r_0 and r_1 and subtract the difference between the BLYP (Refs. 35 and 36) energes with and without the approximation to the inverse (i.e., both calculations use the DF approximation). We also show the result if we do not bump the fitting functions, but merely drop those functions farther than r_1. For this example we use a small value of r_1 (3 a.u.) for clarity. The results are shown in Fig. 1. It is apparent that the unbumped approximate fit is problematic; the energy surface has a discontinuity and there will be a spurious force at this point.

TABLE III. BLYP error per carbon atom using the LADF method, in μH. For each column, the multipoles of each atomic fit were constrained, up to the given value (zero represents charge conserved). The fitting basis sets are of such high quality that constraints reduce the accuracy, though marginally. The basis set used is SV (Ref. 28), with the universal fitting basis (Ref. 29).

Graphite sheet size	No constraints	0	1	2
16	25.8	26.8	28.1	28.8
76	24.4	25.8	27.0	28.4
102	24.0	25.2	25.7	27.0
184	28.7	30.8	31.8	32.5
210	25.2	27.1	26.4	27.4

Presumably, the weakness of this method would be long-ranged interactions. For example, errors in the charge of the fit will have significant influence due to the slow r^{-1} decay of the Coulomb potential. For large systems even small errors may accumulate, especially if the fitting procedure tends to under- or overestimate the charge of all regions. In the normal density fitting method, the fit extends over the entire molecule, so local functions compensate for far-away errors in the density fit.[32] Charge (and further pole) conservation has been accomplished previously for fitting algorithms.[37,38] We will apply these constraints to the local atomic fits, constraining the atomic charges (and higher moments if used) according to the atomic density partitioning based on Eq. (13). This is accomplished using the method of Lagrange multipliers. As shown in Table III, contrary to intuition, a high quality fitting basis set may actually be undermined by imposed constraints. These results are far from sufficient proof of accuracy with system size, so we recommend charge conservation to prevent growth of error with system volume. All calculations in this paper henceforth use charge conservation.

Tables IV–VI show the error for using an approximate inverse on BLYP calculations of linear alkanes, graphite sheets, and diamond chunks, respectively. The basis sets used are a split valence (SV) type, a split valence plus polarization function type (SVP), and a triple zeta (TZ) type.[28] LADF exhibits extremely minor increased error relative to full DF.

TABLE IV. Error (in μH) per carbon for DF and LADF BLYP calculations on alkane chains. The fitting basis set used is the universal fitting basis (Ref. 29).

	Alkane chain length	DF	LADF
SV	10	36.5	41.0
	20	35.9	40.7
	40	35.9	40.8
	120	35.4	40.5
	200	35.3	40.5
SVP	10	44.1	47.3
	20	43.3	46.6
	40	43.6	46.6
	120	42.9	46.0
	200	43.1	46.3
TZ	10	50.8	54.3
	20	55.3	59.7
	40	58.9	64.1
	120	58.3	65.3
	200	60.1	68.0

TABLE V. Error (in μH) per carbon for DF and LADF BLYP calculations on graphite sheets. The fitting basis set used is the universal fitting basis (Ref. 29).

	Graphite sheet size	DF	LADF
SV	16	22.6	26.8
	76	19.1	25.8
	102	18.4	25.2
	184	22.1	30.8
	210	18.4	27.1
SVP	16	28.4	41.0
	76	23.2	41.7
	102	22.9	42.0
	184	22.9	47.2
	210	24.7	49.6
TZ	16	24.3	27.2
	76	24.4	27.0
	102	24.0	26.7
	184	26.0	29.2
	210	23.8	26.5

IV. LADF LINEAR SCALING FORMULATION

Gallant and St.-Amant[39] first performed a linear scaling fit of the density by dividing the system into subunits, with each subunit having its own fit. However, the presence of predefined subunits is undesirable, as the rearrangement of atoms (being central to chemistry) between subunits could produce discontinuities (if the subunits were redefined) or hysteresis (if the subunits are not redefined, and atoms are swapped), for example. The LADF procedure introduced in the previous section provides a natural solution to this problem (a problem which has also been encountered in applying density fitting to local electron correlation methods[40]). In this section we will describe the use of LADF in a linear scaling RI-J method.

In the case of Coulomb matrix construction for a SCF calculation, the interactions we consider are between a function pair, $\mu\nu(r)$, and the density, $\rho(r)$. Weigend provides optimized basis sets[29] for computing such interactions. In Table VII we show the number of fitting functions used for the test systems we calculated. In fact, only a modest number (approximately $3N_{\text{basis}}$) of auxiliary basis functions are required

TABLE VI. Error (in μH) per carbon for DF and LADF BLYP calculations on diamond chunks. The fitting basis set used is the universal fitting basis (Ref. 29).

	Diamond chunk size	DF	LADF
SV	11	31.1	32.0
	87	28.4	34.6
	168	52.5	48.5
	246	34.9	54.3

TABLE VII. The number of fitting functions used for DF and LADF calculations on the test systems calculated in this paper.

Molecule	DF/LADF fitting functions
$C_{10}H_{22}$ (linear)	732
$C_{20}H_{42}$ (linear)	1442
$C_{40}H_{82}$ (linear)	2862
$C_{120}H_{242}$ (linear)	8542
$C_{200}H_{402}$ (linear)	14222
$C_{16}H_{10}$ (graphite)	894
$C_{76}H_{22}$ (graphite)	3966
$C_{102}H_{26}$ (graphite)	5284
$C_{184}H_{34}$ (graphite)	9390
$C_{210}H_{38}$ (graphite)	10708
$C_{11}H_{18}$ (diamond)	737
$C_{87}H_{70}$ (diamond)	5033
$C_{168}H_{106}$ (diamond)	9398
$C_{246}H_{150}$ (diamond)	13704

to fit quite a large number of function pairs, presumably due to significant linear dependence of the function pairs,[41,42] which increases with basis set size.

The initial fitting steps of the algorithm are done only once and the results may be stored for use in the following SCF iterations. Computed approximate inverses are stored in core or on disk for future use. Also, three-center fit integrals are computed and stored in memory. Due to the limited range of the fitting procedure, the number of fitting integrals required is quite small and the onset of linear scaling is rapid.

Given a density (such as from a previous SCF iteration or from an initial "guess"), our procedure loops over each atom of the molecule. Three-center integrals are performed for all auxiliary basis functions near the atom. Per atom, the number of nearby auxiliary basis functions is independent of size, once the system is larger than r_1. The integrals are then transformed by the approximate inverse, yielding fitting coefficients for the density to be used in the second and third terms of Eq. (18),

$$(\rho_{\mathrm{A}}|L) = \sum_{\mu\nu} (\mu\nu|L) P_{\mu\nu,\mathrm{A}}, \tag{19}$$

$$\tilde{\rho}_K = \sum_{\mathrm{A}} \sum_{L} (K|L)^{-1}_{\mathrm{A}} (\rho_{\mathrm{A}}|L). \tag{20}$$

.

The interaction of a fit to $\mu\nu$ with the fit density requires the interaction of an auxiliary function, K, with the fit density, as in the third term of Eq. (18):

$$\tilde{J}_L = (L|K)\tilde{\rho}_K. \tag{21}$$

This is a standard J-type matrix formation, and CFMM can be used to compute it in linear work.

Full three-center Coulomb integrals are needed for two pieces, $\langle\tilde{\mu\nu}|\rho\rangle$ and $\langle\mu\nu|\tilde{\rho}\rangle$, the first and second terms of Eq. (18), respectively,

$$J_L = \sum_{\mu\nu} (L|\mu\nu) P_{\mu\nu}, \tag{22}$$

$$J_{\mu\nu} \leftarrow \sum_{K} (\mu\nu|K)\tilde{\rho}_K. \tag{23}$$

These are also standard J-type matrices, which are computed in linear work using CFMM.

We do not compute a fit of each individual $\mu\nu$. Instead, J_L and $\tilde{J}_L$ are scaled by $(K|L)^{-1}_X$, where X is the source atom of $\mu\nu$. A transformation is performed for each atom's fitting functions. This is equivalent to the procedure for the density,

$$I_{K,\mathrm{X}} = \sum_{L} (K|L)^{-1}_{\mathrm{X}} (J_L - \tilde{J}_L), \tag{24}$$

$$J_{\mu\nu} \leftarrow \sum_{\mathrm{X}} \sum_{K\in\mathrm{X}} (\mu\nu|K) I_{K,\mathrm{X}}, \tag{25}$$

where $I_{K,\mathrm{X}}$ is comprised of those auxiliary basis functions used by atom X.

The pivotal difference between this and the standard density fitting procedure is that the full inverse matrix is not needed. Instead, we require one Cholesky decomposition for every atom, though of a small, fixed size. The reduced memory usage of the algorithm is important. Storage of the full auxiliary basis interaction matrix requires N^2_{aux} memory. As N_{aux} is significantly larger than the number of basis functions, storing a single auxiliary basis matrix may limit the feasibility for large systems. Also, linear dependence in the auxiliary interaction matrix is possible for high quality fitting basis sets of condensed systems. In this case, Cholesky decomposition may not be available, and an extremely expensive singular value decomposition (SVD) may be necessary. Indeed, in the case of the diamond chunk $C_{168}H_{106}$, the LAPACK (Ref. 43) routines dpotrf/dpotri fail and SVD is necessary.

All calculations in this paper were run on one processor of a 2.3 GHz Xserve G5, using up to 8 Gbytes of RAM. SCF convergence criteria were set to 10^{-6}. Thresholds for discarding shell pairs were set to 10^{-8}. In the case of timing J-matrix computation, the per-iteration time was averaged over the SCF iterations, due to screening of the density matrix.

Problems may arise if a large degree of linear dependence is present in the primary basis set. This is presumably because the atomic division of the density is unnatural in this case, and the resulting atomic densities are not fitted well. A vicious cycle then ensues, which renders the density even less well fit. The addition of extra fitting functions, increasing the fitting radius, or loosening the linear dependence threshold are obvious solutions to this deficiency of the method. For the following calculations on linear alkanes a linear dependence threshold of 10^{-4} was used. For the graphite sheets and diamond chunks, a value of 10^{-3} was used.

The LADF/ARI method has been implemented into a development version of the quantum chemistry package Q-CHEM.[44]

We tested the algorithm on a series of linear alkanes, up to $C_{200}H_{402}$, hydrogen-terminated graphite sheets up to $C_{210}H_{38}$, and diamond chunks up to $C_{246}H_{150}$ (see Fig. 2). Whereas inversion of the entire auxiliary interaction matrix took 738 s for $C_{200}H_{402}$, the approximated inverse was computed in only 24 s. For $C_{210}H_{38}$, complete inversion takes

TABLE VIII. CPU timing per iteration (in seconds) for Coulomb calculations on alkane chains using standard exact methods, traditional DF, and LADF. Here "Exact" refers to the combination of CFMM and J-engine, without fitting methods.

	Alkane chain length	Exact	DF	LADF
SV	10	1.9	1.4	1.6
	20	4.9	3.2	3.5
	40	13.4	6.9	7.3
	120	57.5	23.1	23.4
	200	108.5	45.1	41.7
SVP	10	3.2	1.8	1.9
	20	8.3	4.1	4.3
	40	21.3	8.8	9.0
	120	89.6	30.9	29.5
	200	162.7	67.7	52.5
TZ	10	6.2	2.3	2.2
	20	20.4	5.5	5.0
	40	56.5	11.8	10.6
	120	253.3	43.1	35.9
	200	509.3	77.7	63.5

353 s, but approximate inversion only takes 33 s. For the diamond chunk $C_{246}H_{150}$, full inversion took 636 s, while approximate inversion took 143 s. The advantages of approximate inversion lessen for higher dimensional molecules (all else being equal), where inversion regions contain more atoms. For larger systems, however, the advantages of a true linear scaling algorithm grow inexorably.

Near-field (NF) three-center integrals must be performed each iteration and are the bottleneck step. Both three- and two-center works utilize CFMM techniques, which take a small fraction of the total time. Further timing improvement could be obtained by optimally partitioning near- and far-field work, which we have not reoptimized for DF. We take advantage of the J-engine[25] of Shao *et al.* for near field three-center work and observe approximately a factor of 2 speedup. Three-center integrals are also required for the initial fitting step. In our implementation, fitting theree-center

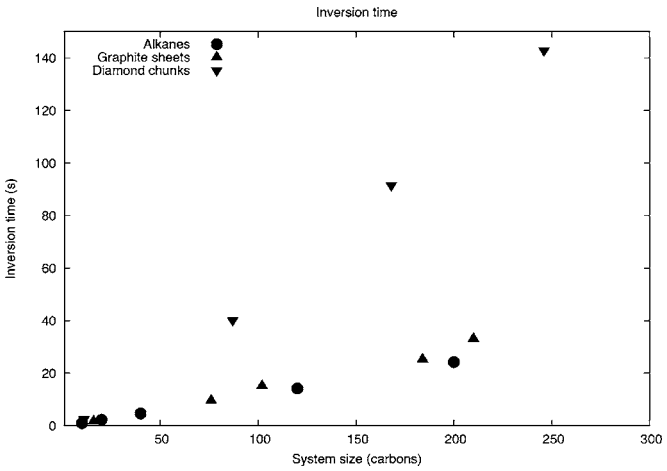

FIG. 2. The total CPU time for LADF fitting inversion for a series of linear alkanes, graphite sheets, and diamond chunks. The fitting basis set is Weigend's universal fitting basis.

TABLE IX. CPU timing per iteration (in seconds) for Coulomb calculations on graphite sheets using standard exact methods, traditional DF, and LADF.

	Graphite sheet size	Exact	DF	LADF
SV	16	4.8	1.7	2.2
	76	91.2	21.6	19.8
	102	134.5	32.1	27.5
	184	355.7	63.9	54.7
	210	438.8	77.7	64.8
SVP	16	8.78	2.3	3.1
	76	137.2	29.5	27.3
	102	257.7	44.0	39.9
	184	660.9	88.8	77.4
	210	705.8	109.9	91.9
TZ	16	18.9	3.4	3.5
	76	476.9	43.4	33.2
	102	781.4	68.3	48.3
	184	1888.7	135.6	99.3
	210	2489.7	158.3	128.2

integrals are not optimized as effectively[45] as the per-iteration pass, and so run approximately three times slower than the per-iteration near-field work, but must be performed only once per SCF calculation. Timing for alkane chains, graphite sheets, and diamond chunks are presented in Tables VIII–X, respectively. Although with LADF and DF we surpass the linear scaling combination of CFMM and J-engine, with more diffuse basis sets a comparison should be made with FTC, which handles these types of functions very efficiently.

LADF and DF have approximately the same per-iteration cost. The bottleneck near-field step, J-engine evaluation, is the same for both methods. LADF requires half as many fundamental integral computations, but requires more overhead, effectively making the cost equivalent. Speedups relative to exact integral evaluation (using CFMM and the J-engine) are approximately a factor of 10.

V. PERFORMANCE ASSESSMENT

Referring to Table I, it is clear that for large enough systems, inversion of the auxiliary basis interaction matrix will dominate the cost of building the *J* matrix. Our example, linear alkanes, is a best case for LADF, because there are relatively few near-field integrals to evaluate, yet the full inversion process of DF is as expensive as if the system were

TABLE X. CPU timing per iteration (in seconds) for Coulomb calculations on diamond chunks using standard exact methods, traditional DF, and LADF. The basis set used is SV. It was necessary to use the result of the LADF calculation as a guess for the exact calculation. In this case, the exact calculations did not benefit from density matrix thresholding, and in practice would require approximately half the CPU time.

Diamond chunk size	Exact	DF	LADF
11	4.3	1.5	1.4
87	1545.8	131.3	73.3
168	6804.9	400.2	232.7
246	15400.6	569.3	411.6

TABLE XI. A breakdown of the steps of DF and LADF Coulomb calculations on graphite sheets. Steps labeled "Fitting" indicate matrix multiplication using the auxiliary basis interaction inverse, not calculation of the inverse. The normal basis set used is SVP, and the auxiliary basis set is the universal fitting basis.

Carbons	NF (DF)	NF (LADF)	CFMM (DF)	CFMM (LADF)	Fitting (DF)	Fitting (LADF)	Two-center (LADF)
16	2.22	1.70	0	0.37	0.01	0.25	0.23
76	25.50	15.70	3.31	3.39	0.31	2.38	2.08
102	38.37	23.76	4.31	4.31	0.51	3.24	3.04
184	72.95	44.14	9.18	9.34	0.91	6.34	5.61
210	86.28	53.06	10.75	10.89	1.02	7.26	6.53

globular. For two- or three-dimensional molecules, the crossover between inversion and the CFMM/integral steps will have many more atoms (≈350 carbons for graphite, ≈700 carbons for diamond, for the SV basis set with the universal basis set). However, for LADF there is extra computational cost in doing the initial fit. This extra work is small compared with the cost of full inversion for large systems and could run faster in a fully optimized[45] code.

It is worthwhile to note that DF and LADF have approximately the same cost of building the *J* matrix, once the inversion has been performed. In Table XI we give the timings for each part of the *J* matrix cycle for graphite sheets using the SVP basis set, for both DF and LADF. Even though LADF has one near field pass to DF's two, LADF's near field does not run twice as fast. The equalizing factor is the J-engine, as discussed previously. The near-field speedup of LADF is somewhat offset by the overhead associated with fitting each atom separately and with the additional two-center step that is part of the Dunlap correction. We note that the two-center step does use CFMM.

We attempt to summarize the total benefits of LADF in Table XII. For small systems, DF is somewhat faster due to it having less overhead. For larger systems, LADF benefits from performing fewer near-field integrals, as well as a much reduced cost for inversion. To compare the two in a balanced way, we have multiplied the per-iteration cost by 10, to simulate a typical SCF calculation (although SCF calculations can take many more iterations). $C_{168}H_{106}$ has been omitted from the comparison because the optimized Cholesky decomposition failed, due to near-zero eigenvalues. Linear systems benefit much more from LADF, since inversion is a much greater percentage of the cost. For the three-dimensional diamond chunks, the cost of DF inversion was comparable to the cost of the extra LADF fitting integrals, though eventually inversion would dominate.

TABLE XII. Total LADF *J*-build time as a percentage of DF *J*-build time. This includes the cost of inversion, calculating fitting integrals, and building ten *J* matrices.

Molecule	SV	SVP	TZ
$C_{10}H_{22}$ (linear)	125.0%	121.0%	104.9%
$C_{20}H_{42}$ (linear)	116.8%	115.1%	98.9%
$C_{40}H_{82}$ (linear)	103.7%	104.4%	93.4%
$C_{120}H_{242}$ (linear)	65.0%	69.0%	67.1%
$C_{200}H_{402}$ (linear)	39.8%	42.1%	47.7%
$C_{16}H_{10}$ (graphite)	143.4%	153.2%	117.2%
$C_{76}H_{22}$ (graphite)	89.4%	95.5%	82.0%
$C_{102}H_{26}$ (graphite)	77.9%	87.4%	73.5%
$C_{184}H_{34}$ (graphite)	74.3%	81.6%	74.9%
$C_{210}H_{38}$ (graphite)	67.3%	75.2%	77.1%
$C_{11}H_{18}$ (diamond)	115.2%		
$C_{87}H_{70}$ (diamond)	62.8%		
$C_{246}H_{150}$ (diamond)	74.3%		

VI. CONCLUSIONS

We have updated the RI/DF procedure to use CFMM, which makes the interaction step of DF linear scaling. Furthermore, we demonstrated that using the J-engine offers significant speedups insofar as DF no longer requires two complete computations of near-field integrals. Finally, we have shown that the DF procedure can be modified to be linear scaling, while retaining acceptable accuracy (error is increased marginally over DF for carbon systems). With our linear scaling algorithm, the time required to compute fitting coefficients scales linearly with system size and is negligible compared with the overall cost of a SCF iteration. The linear scaling fit is achieved using a "bump function," which sets the contribution of a far-away fitting function to zero so that it may not be considered in the fit of a local piece of the density. These results demonstrate that the known advantages of auxiliary basis methods for Coulomb builds in medium-sized molecules can be extended to very large systems.

ACKNOWLEDGMENTS

One of the authors (A.S.) would like to thank Yihan Shao for helpful discussions regarding the use of the J-engine in this work. This work was supported by a subcontract from Q-Chem Inc. based on a NIH SBIR award, with additional support from the National Science Foundation (CHE-0535710). Another author (M.H.G.) is a part owner of Q-Chem Inc.

[1] A. Szabo and N. Ostlund, *Modern Quantum Chemistry* (Dover, Mineola, NY, 1996).
[2] R. Parr and W. Yang, *Density-Functional Theory of Atoms and Molecules* (Oxford University Press, New York, 1989).
[3] E. Schwegler and M. Challacombe, J. Chem. Phys. **105**, 2726 (1996).
[4] C. Ochsenfeld, C. White, and M. Head-Gordon, J. Chem. Phys. **109**, 1663 (1998).
[5] R. E. Stratmann, G. Scuseria, and M. Frisch, Chem. Phys. Lett. **257**, 213 (1996).
[6] L. Greengard, Science **265**, 909 (1994).
[7] C. A. White, B. G. Johnson, P. M. W. Gill, and M. Head-Gordon, Chem. Phys. Lett. **230**, 8 (1994).
[8] M. C. Strain, G. E. Scuseria, and M. J. Frisch, Science **271**, 51 (1996).
[9] R. B. Murphy, Y. Cao, M. D. Beachy, M. N. Ringnalda, and R. A. Friesner, J. Chem. Phys. **112**, 10131 (2000).

[10] M. Sierka, A. Hogekamp, and R. Ahlrichs, J. Chem. Phys. **118**, 9136 (2003).
[11] G. Lippert, J. Hutter, and M. Parrinello, Theor. Chem. Acc. **103**, 124 (1999).
[12] L. Füsti-Molnár and P. Pulay, J. Chem. Phys. **117**, 7827 (2002).
[13] L. Füsti-Molnár and J. Kong, J. Chem. Phys. **122**, 074108 (2005).
[14] A. L. Sklar, J. Chem. Phys. **7**, 984 (1939).
[15] R. Mulliken, J. Chem. Phys. **46**, 497 (1949).
[16] P. Löwdin, J. Chem. Phys. **21**, 374 (1953).
[17] F. P. Billingsley and J. E. Bloor, J. Chem. Phys. **55**, 5178 (1971).
[18] J. L. Whitten, J. Chem. Phys. **58**, 4496 (1973).
[19] B. I. Dunlap, J. W. D. Connolly, and J. R. Sabin, J. Chem. Phys. **71**, 3396 (1979).
[20] B. Dunlap, THEOCHEM **501–502**, 221 (2000).
[21] M. D. Newton, J. Chem. Phys. **51**, 3917 (1969).
[22] E. J. Baerands, D. E. Ellis, and P. Ros, Chem. Phys. **2**, 41 (1973).
[23] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery, *Numerical Recipes in C*, 2nd ed. (Cambridge University Press, Cambridge, U.K., 1999).
[24] S. Goedecker, Rev. Mod. Phys. **71**, 1085 (1999).
[25] Y. Shao and M. Head-Gordon, Chem. Phys. Lett. **323**, 425 (2000).
[26] C. White and M. Head-Gordon, J. Chem. Phys. **104**, 2620 (1996).
[27] G. Ahmadi and J. Almlöf, Chem. Phys. Lett. **246**, 364 (1995).
[28] A. Schäfer, H. Horn, and R. Ahlrichs, J. Chem. Phys. **97**, 2571 (1992).
[29] F. Weigend, Phys. Chem. Chem. Phys. **8**, 1057 (2006).
[30] O. Vahtras, J. Almlöf, and M. W. Feyereisen, Chem. Phys. Lett. **213**, 514 (1993).
[31] B. I. Dunlap, THEOCHEM **529**, 37 (2000).
[32] Y. Jung, A. Sodt, P. M. W. Gill, and M. Head-Gordon, Proc. Natl. Acad. Sci. U.S.A. **102**, 6692 (2005).
[33] M. Benzi and C. Meyer, SIAM J. Sci. Comput. (USA) **16**, 1159 (1995).
[34] C. Fonseca Guerra, J. G. Snijders, G. te Velde, and E. J. Baerends, Theor. Chem. Acc. **99**, 391 (1998).
[35] A. Becke, J. Chem. Phys. **96**, 2155 (1992).
[36] C. Lee, W. Yang, and R. G. Parr, Phys. Rev. B **37**, 785 (1988).
[37] C. Van Alsenoy, J. Comput. Chem. **9**, 620 (1988).
[38] D. Wilhite and R. Euwema, Chem. Phys. Lett. **20**, 610 (1973).
[39] R. T. Gallant and A. St-Amant, Chem. Phys. Lett. **256**, 569 (1996).
[40] H.-J. Werner, F. Manby, and P. Knowles, J. Chem. Phys. **118**, 8149 (2003).
[41] N. H. F. Beebe and J. Linderberg, Int. J. Quantum Chem. **12**, 683 (1977).
[42] D. W. O'Neal and J. Simons, Int. J. Quantum Chem. **36**, 673 (1989).
[43] E. Anderson, Z. Bai, C. Bischof *et al.*, LAPACK *Users' Guide*, 3rd ed. (Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999).
[44] Y. Shao, L. Füsti-Molnár, Y. Jung *et al.*, Phys. Chem. Chem. Phys. **8**, 3172 (2006).
[45] R. Ahlrichs, Phys. Chem. Chem. Phys. **6**, 5119 (2004).