Linear Scaling Computation of the Fock Matrix. V. Hierarchical Cubature for Numerical Integration of the Exchange-Correlation Matrix. [§††]

Matt Challacombe[†]
Theoretical Division
Group T-12, MS B268
Los Alamos National Laboratory
Los Alamos, NM 87545

(Dated: August 16, 2000)

Hierarchical Cubature is a new method for achieving linear scaling computation of the exchange-correlation matrix central to Density Functional Theory. Hierarchical Cubature combines a k-dimensional generalization of the binary search tree with adaptive numerical integration involving an entirely Cartesian grid. Hierarchical Cubature overcomes strong variations in the electron density associated with nuclear cusps through multi-resolution rather than spherical-polar coordinate transformations. This unique Cartesian representation allows use of the exact integration error during grid construction, supporting $\mathcal{O}(\log N)$ range-queries that exploit locality of the Cartesian Gaussian based electron density. Convergence is controlled by τ_r, which bounds local integration error of the electron density.

An early onset of linear scaling is observed for RB3LYP/6-31G** calculations on water clusters, commencing at $(H_2O)_{30}$ and persisting with decreasing values of τ_r. Comparison with nuclear weight schemes suggests that the new method is competitive on the basis of grid points per atom. Systematic convergence of the RPBE0/6-31G** Ne_2 binding curve is demonstrated with respect to τ_r.

I. INTRODUCTION

Hybrid Hartree-Fock/Density Functional Theory (HF/DFT)[1–5] has proven to be an important recent development in electronic structure theory, with a high accuracy to cost ratio that has dramatically extended the size and reliability of *ab initio* calculations in many new areas. Hybrid HF/DFT methods are economical because they are Self-Consistent-Field (SCF) or independent particle models, where the fundamental variable is the density matrix $\mathbf{P}$. Recently, $\mathcal{O}(N)$ methods for solving various aspects of SCF theory have been introduced that enjoy a computational cost scaling only linearly with system size. While the promise of these methods for performing large scale HF/DFT applications is real, programs able to achieve full linear scaling for HF/DFT have not been forthcoming, due perhaps to a number of difficult bottlenecks that must be overcome simultaneously.

In HF/DFT, the Fock matrix is

$$\mathbf{F} = \mathbf{h} + \mathbf{J} + a_0\,\mathbf{K}_x^{\rm HF} + \mathbf{K}_{xc}^{\rm DFT}\,, \qquad (1)$$

where $\mathbf{h}$ is the core Hamiltonian, $\mathbf{J}$ is the Coulomb or Hartree matrix, $a_0 \sim 0.20$ is a mixing parameter for the exact Hartree-Fock exchange matrix $\mathbf{K}_x^{\rm HF}$, and $\mathbf{K}_{xc}^{\rm DFT}$ is the exchange-correlation matrix. Typically $\mathbf{K}_{xc}^{\rm DFT}$ is derived from an exchange-correlation functional in the generalized gradient approximation (GGA), $\mathbb{E}_{xc}^{\rm GGA}[\rho]$, with a_0 of the DFT exchange excluded[1–5].

Due to inefficiencies associated with the implementation of Hartree-Fock exchange in plane wave, grid, wavelet, Slater and numerical orbital methods, the only viable approach for large scale HF/DFT calculations appears to involve the use of standard quantum chemical basis sets that employ the Cartesian Gaussian, Linear Combination of Atomic Orbital (CG-LCAO) approximation[6–9], for which well developed analytic methods exist to evaluate the relevant two-electron integrals[10].

There are at least four major bottlenecks that must be overcome to achieve linear scaling HF/DFT (three for computing $\mathbf{F}$, and one for solving the SCF equations). In this series[11–14], linear scaling methods for computing CG-LCAO based $\mathbf{J}$ and $\mathbf{K}_x^{\rm HF}$ matrices have been developed, and accuracies comparable to direct SCF[15,16] have been demonstrated for three-dimensional systems. Also, bottlenecks associated with Löwdin orthogonalization[17,18] and Roothann-Hall[19,20] solution of the SCF equations have recently been overcome with sparse-blocked matrix technologies and density matrix minimization[21,22], likewise demonstrating linear scaling and error control for three-dimensional benchmark suites. On the other hand, methods for linear scaling com-

§ Cite this preprint as LA-UR-00-1618.
†† Submitted to the Journal of Chemical Physics.

putation of the CG-LCAO exchange-correlation matrix $\mathbf{K}_{xc}^{\mathrm{DFT}}$[23,24] are less well developed, achieving linear scaling only for small basis sets and extremely large or low dimensional systems[25].

Products of the CG-LCAO basis functions, $\rho_{ab}(\mathbf{r}) = \phi_a(\mathbf{r})\,\phi_b(\mathbf{r})$, are the fundamental participants in formation of the exchange-correlation matrix. In particular, modern algorithms compute the exchange-correlation matrix with

$$K_{ab}^{xc} = \int d\mathbf{r} \left[\frac{\partial \mathbb{E}_{xc}}{\partial \rho}\, \rho_{ab} - 2 \frac{\partial \mathbb{E}_{xc}}{\partial |\boldsymbol{\nabla}\rho|^2}\, \boldsymbol{\nabla}\rho_{ab} \cdot \boldsymbol{\nabla}\rho \right] , \quad (2)$$

where restricted SCF theory has been assumed and the second term disappears in the local density approximation[26]. Because the exchange-correlation functional $\mathbb{E}_{xc}[\rho]$ is in general non-analytic, the corresponding matrix elements must be computed numerically.

CG-LCAO basis functions are typically contractions of primitive Gaussian functions: $\phi_a(\mathbf{r}) = \sum_i C_i^a\, \varphi_i(\mathbf{r})$. Both CG-LCAO basis function products,

$$\begin{aligned} \rho_{ab}(\mathbf{r}) &= \phi_a(\mathbf{r})\,\phi_b(\mathbf{r}) & (3) \\ &= \sum_p \sum_{LMN} c_{p,LMN} \frac{\partial^{L+M+N}}{\partial P_x^L \partial P_y^M \partial P_z^N}\, e^{-\zeta_p(\mathbf{r}-\mathbf{P})^2} \\ &= \sum_p \varrho_p(\mathbf{r}) \,, \end{aligned}$$

and the density,

$$\begin{aligned} \rho(\mathbf{r}) &= \sum_{ab} P_{ab}\, \phi_a(\mathbf{r})\,\phi_b(\mathbf{r}) & (4) \\ &= \sum_q \sum_{LMN} d_{q,LMN} \frac{\partial^{L+M+N}}{\partial Q_x^L \partial Q_y^M \partial Q_z^N}\, e^{-\zeta_q(\mathbf{r}-\mathbf{Q})^2} \\ &= \sum_q \varrho_q(\mathbf{r}) \,, \end{aligned}$$

may be resolved into *primitive distributions* ϱ, which are linear combinations of Hermite-Gaussian functions involving the same "mother" Gaussian with scale factor ζ and center $\mathbf{Q}$[27–29]. For typical CG-LCAO basis functions, ζ spans $[10^{-2}, 10^5]$. Also, due to the Gaussian product theorem the coefficients $\mathbf{c}_p$ and $\mathbf{d}_q$ decay as Gaussians with separation of the product basis function centers, typically spanning $[10^{-12}, 10^2]$[27–29]. Thus, extreme variability in both the *range* and the *magnitude* of the ϱ suggest a multi-scale approach to computation of $\mathbf{K}_{xc}^{\mathrm{DFT}}$.

The strongly peaked nature of the electron density about nuclear centers requires special treatment, and has inspired sophisticated algorithms for domain decomposition and variable transformation, yielding integrations schemes with a minimal number of grid points per atom $\langle N_{\mathrm{Pts}} \rangle$. There are at least three main decomposition paradigms; nuclear weight functions[30–38], Voronoi polyhedra[39], and parallelepipeds[40,41]. These methods all employ a transformation to spherical coordinates, with the Jacobian suppressing nuclear cusps in the electron density[42]. However, these cusps are not strong singularities, and it is interesting to consider an adaptive and purely Cartesian approach to numerical integration, where subdivision achieves arbitrary resolution of strongly peaked domains while sparing effort in regions that are slowly varying. Such a scheme is appealing in the development of linear scaling methods as it suggests a "fast" tree-based approach.

This concept is pursued to fruition, and a new linear scaling method for computing the exchange-correlation matrix is presented that fully exploits the muti-scale structure of ρ and ρ_{ab} by avoiding work at the level of primitive distributions. This is accomplished through hierarchical representation of both the density and the integration grid using the *k-d tree* data structure[43–45], allowing efficient *orthogonal range queries* to dynamically select only numerically significant contributions. The *k-d tree* data structure is used to implement a decomposition paradigm that employs recursive bisection and non-product Cartesian Gaussian integration rules (C_3 rules), resulting in a Hierarchical Cubature able to resolve the strong variations and multiple length scales encountered in numerical integration of the exchange-correlation matrix.

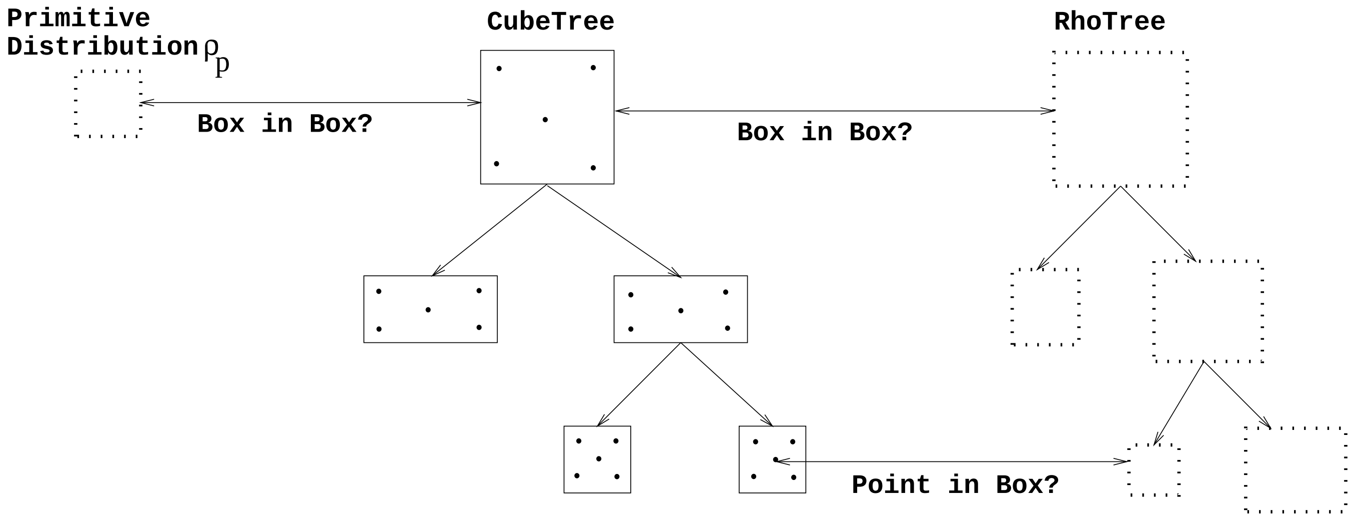
II. HIERARCHICAL CUBATURE

Hierarchical Cubature (HiCu) employs the *k-dimensional binary search tree* (*k-d tree*)[43–45] data structure to represent and evaluate both the electron density and the integration grid. The *k-d tree* associated with the density is the *RhoTree*, and the *k-d tree* associated with the grid is the *CubeTree*. The *k-d tree* is a k-dimensional generalization of the binary search tree, allowing rapid location and evaluation of minimally overlapping grid and density components.

Each node in a *k-d tree* includes a k-dimensional hyper-rectangle or bounding box (`BBox`) that encloses the spatial data contained by it and its children. A range query is a search of the *k-d tree* for all data within a given Euclidean distance. Performing a range query involves starting at the root of a *k-d tree*, and recursively searching for `BBox`es that overlap a point (or a `BBox` about the point, depending on how the search is formulated). Recursive descent of the *k-d tree* is halted when a non-overlapping `BBox` is encountered, while those nodes with overlap are traversed until all leaf nodes within range have been accessed, resulting in an $\mathcal{O}(\log_2 N)$ search[43–45].

Two *k-d tree* structures are employed by HiCu, the RhoTree and the CubeTree, are shown in Fig. 1. In the case of the RhoTree, the point-`BBox` implementation is used, wherein the range is incorporated into the RhoTree by expanding each `BBox` by a certain *extent*. Alternatively, the interaction of the CubeTree with a primitive distributions ϱ_p is formulated in terms of `BBox`–`BBox`

FIG. 1: Use of the *k-d tree* in computation of the exchange-correlation matrix. In construction of the CubeTree, locality of the density is fully exploited by evaluating only those primitive components that contribute to a grid point. This is accomplished through *range queries*, which involve walking on the RhoTree performing a check of each node for significance, such that children of insignificant nodes are not traversed. At a higher level, this involves a test for `BBox`–`BBox` intersection, and for `BBox`es that do overlap, examining the RhoTree children for those that contain a grid point. In evaluation of the exchange-correlation matrix elements, locality of the primitive distributions ϱ_p is exploited through range queries that involve walking on the CubeTree, testing for `BBox`–`BBox` intersection.



overlaps.

A The RhoTree

The electron density is represented by the 5-dimensional RhoTree. The five coordinates include three Cartesian directions corresponding to the distribution centers $\mathbf{Q}$, the distribution amplitude, and the distribution scale factors. The RhoTree is constructed by recursive bisection starting at the root, which encompasses the entire density. With each subdivision, a new 3-dimensional `BBox` containing the centers $\mathbf{Q}$ is generated, and a new maximum distribution amplitude

$$\texttt{MaxAmp} = \max_{q \in \texttt{BBox}} \sqrt{\mathbf{d}_q \cdot \mathbf{d}_q}\,, \tag{5}$$

and minimum distribution exponent

$$\texttt{ZetaMin} = \min_{q \in \texttt{BBox}} \zeta_q\,, \tag{6}$$

are determined. Bisection proceeds until each distribution is resolved into a leaf node. This is shown in Fig. 2.

Given a target error τ_ρ in the density, each `BBox` in the RhoTree is expanded by the amount

$$\texttt{Extent} = \sqrt{\frac{|\log\left(\texttt{MaxAmp}/\tau_\rho\right)|}{\texttt{ZetaMin}}} \tag{7}$$

which is unique to each node. Thus, the contribution a node makes outside its `BBox` is assumed to be zero. It is now possible to evaluate the density at a given point by starting at the root of the RhoTree and traversing only those nodes having a `BBox` containing the point. This is efficient because the test for point-box overlap is cheap, because the search is $\mathcal{O}(\log_2 N)$, and because exclusion of significant density components occurs early in traversal as a result of including ordinates for the amplitude and scale factor.

B The CubeTree

The CubeTree is constructed through recursive bisection of a Cartesian domain enclosing the density, applying a fixed fully symmetric C_3 (cubature) rule[46–49] to nodes with increasingly small volumes in areas where the integrand varies rapidly. This is similar to conventional methods for the adaptive solution of multiple integrals[50–52], which employ local error estimates based on sequences of *embedded* grids[53–55], where rules of different degree are constrained to employ common points.

The differences between HiCu and conventional methods of adaptive integration are (1) the *k-d tree* structure is persistent, with each node of the CubeTree containing a `BBox`; (2) the exact error

$$\texttt{Error} = \left| \int_{\texttt{BBox}} d\mathbf{r}\, \rho(\mathbf{r}) - \sum_{i=1}^{\texttt{NGrid}} \texttt{Wght}[i]\, \rho\left(\texttt{Grid}[i]\right) \right| \tag{8}$$

is used instead of an estimate, allowing freedom in the choice of a cubature rule.

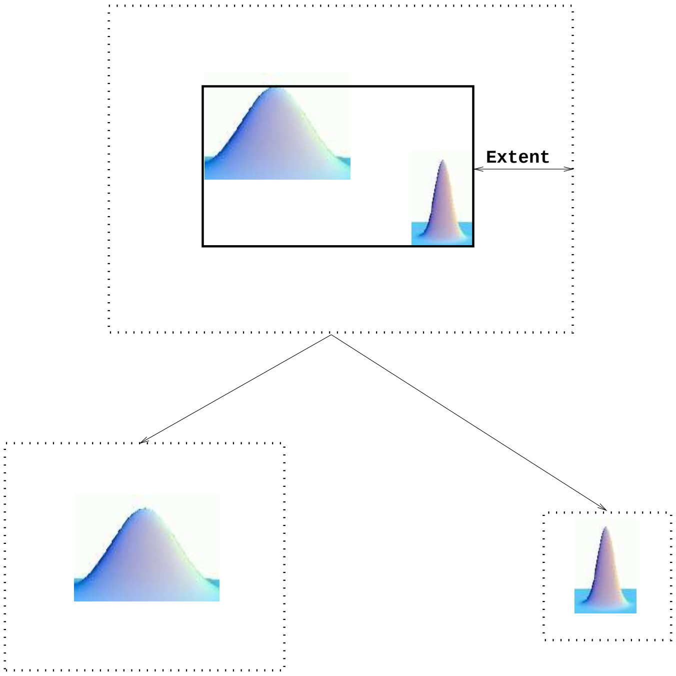


FIG. 2: A terminal branch in the RhoTree. The branch node involves a `BBox` that encloses the centers $\mathbf{Q}$ of two primitive Gaussian distributions (solid lines) in a minimal way. This minimal `BBox` is expanded by an amount `Extent`, given by Eq. (7), to create an expanded `BBox`, shown with dashes. Outside the expanded `BBox`, contributions from the distributions are less than τ_ρ. The branch node is split into two leaf nodes, each containing one primitive distribution with a unique `Extent` and `BBox`.

The accuracy of the HiCu grid is determined by a local error threshold $\tau_\square$; the CubeTree is extended recursively until the error in each leaf node, `Error`, is bounded by $\tau_\square$. This approach improves the grid systematically with respect to the electron count $N_{\rm El}$, but it is not guaranteed to improve with respect to the exchange-correlation energy,

$$E_{xc} = \int d\mathbf{r}\; \mathbb{E}_{xc}\left[\rho(\mathbf{r})\right] \tag{9}$$

or

$$\begin{aligned} G_{\rm xc} &= \sum_{ab} P_{ab} K^{xc}_{ab} \qquad\qquad (10)\\ &= \int d\mathbf{r} \left[\frac{\partial \mathbb{E}_{xc}}{\partial \rho}\, \rho - 2 \frac{\partial \mathbb{E}_{xc}}{\partial |\boldsymbol{\nabla}\rho|^2}\, |\boldsymbol{\nabla}\rho|^2 \right], \end{aligned}$$

which is a global measure of $\mathbf{K}^{\rm DFT}_{xc}$. However, except for very loose thresholds, $\tau_\square \gtrsim 10^{-4}$, the relative errors in $E_{\rm xc}$ and $G_{\rm xc}$ are found to be of the same order as the relative error in $N_{\rm El}$. Others have come to similar conclusions, with the relative error ϵ_r in $N_{\rm El}$ taken as a measure of grid accuracy in a number of recent studies[32,33,35,38].

Recursive construction of the CubeTree starts at the root with a `BBox` that integrates the density to within a requested relative accuracy. For each node (`Cube`) in the CubeTree, the RhoTree is traversed, performing a simple test for `BBox`–`BBox` intersection. For overlapping `BBox`es, contributions to the analytic integral over a `Cube`'s `BBox` are accumulated and evaluation of the density proceeds at the level of grid points, recurring down the RhoTree testing for individual point-`BBox` overlaps. This is shown in Fig. 1.

C Evaluation of $\mathbf{K}_{xc}$

Elements of the exchange-correlation matrix are computed using Eq (2) at the level of primitive distributions, ϱ_p. Each ϱ_p involves a different `Extent`, which determines a unique `BBox` via a straightforward generalization of Eq. (7). Primitive matrix elements are computed by traversing the CubeTree, checking for `BBox`-`BBox` overlap as shown in Fig. 1.

III. IMPLEMENTATION

A Cubature rules

A number of non-product, fully symmetric C_3 rules have been evaluated, with the 14 point, degree 5 rule of Hammer and Stroud[46] being the most efficient, yielding the lowest number of grid points per atom, $\langle N_{\rm Pts}\rangle$. This 14 point rule is used throughout.

B Thresholds

Recursive construction of the CubeTree starts at the root with a `BBox` that analytically integrates the density to within $\epsilon_r = \tau_r * 10^{-1}$, where τ_r is the target relative error in $N_{\rm El}$. This target value is coupled to the local error threshold $\tau_\square$ by $\tau_\square = \delta_r \tau_r$. The value $\delta_r = 0.2$ was found to work well for the systems and basis sets used here, yielding $\epsilon_r \lesssim \tau_r$.

Since the HiCu grid is rebuilt each SCF cycle, it is inefficient to use full precision of the density early in construction of the CubeTree. Therefore, the CubeTree is constructed iteratively, using progressively tighter values of τ_ρ, which controls accuracy of the density according to Eq. (7). In practice, setting $\tau_\rho = 10^{-3}\tau_\square$ yields results that are indistinguishable from those obtained with the full density. With every iteration, $\tau_\square$ is decreased toward its target value, each `BBox` of the RhoTree is re-expanded, and the HiCu grid is refined. Starting at $\tau_\square = 100\ \tau_r$ and progressing to $\tau_\square = \delta_r \tau_r$ in 50 iterations yields significant speed ups.

FIG. 3: The `Cube` object. `Left` is a link to the left child, and `Right` is a link to the right child, or the next node to traverse in the doubly linked list. `Leaf` is a logical flag indicating either branch or leaf-node status. `IXact` is the exact value of the density, integrated analytically over `BBox`. `Grid`, `Wght`, and `Vals` are allocatable, and hold the grid points (3,`NGrid`), the weights (`NGrid`), and the values $\frac{\partial \mathbb{E}_{xc}}{\partial \rho}$, $\frac{\partial \mathbb{E}_{xc}}{\partial |\boldsymbol{\nabla}\rho|^2}$, and $\boldsymbol{\nabla}\rho$ (5,`NGrid`). During grid construction these arrays are deallocated when a node is split.

```
TYPE CubeNode
   LOGICAL                                      :: Leaf
   REAL(DOUBLE)                                 :: IXact
   TYPE(BBox)                                   :: Box
   TYPE(CubeNode),POINTER                       :: Left
   TYPE(CubeNode),POINTER                       :: Right
   REAL(DOUBLE),   POINTER,DIMENSION(:,:)      :: Grid
   REAL(DOUBLE),   POINTER,DIMENSION(:)        :: Wght
   REAL(DOUBLE),   POINTER,DIMENSION(:,:)      :: Vals
END TYPE
```

Matrix elements are computed for atom pairs (A, B) with $\zeta_{\min} |\mathbf{A} - \mathbf{B}|^2 < -\log(10^{-3}\tau_r)$, and entered blockwise into the BCSR sparse matrix representation of $\mathbf{K}_{xc}^{\mathrm{DFT}}$[21,22].

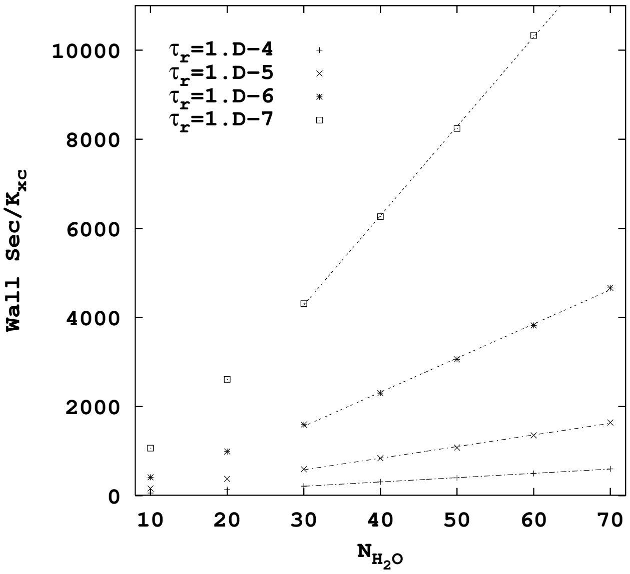
To summarize, a target accuracy τ_r is specified, determining the local absolute cubature error $\tau_{\square}$. The HiCu grid is constructed iteratively using progressively more accurate values of the density. At maximum resolution, each leaf in the CubeTree has achieved an absolute error less than $\tau_{\square} = \delta_r \tau_r$ using numerical values of the density that are accurate to within $\tau_\rho = 10^{-3}\tau_{\square}$.

C Implementation

Program HiCu has been coded in Fortran90 using object oriented principles, and implemented within the MondoSCF suite of linear scaling SCF programs[56]. Both the RhoTree and the CubeTree are implemented as doubly linked lists, using pointers as shown in Fig. 3 for the CubeTree. This allows dynamic memory allocation and tree traversal with recursive subroutines and functions. It is worth pointing out that significant performance improvements were obtained by minimizing argument lists in recursive calls. This approach has resulted in a very compact and efficient code, in contrast with the integer arrays used to implement doubly linked lists in the Fortran77 version of the Quantum Chemical Tree Code[11,28,29].

Evaluation of the density relies heavily on the exponential function `EXP`, while analytic integration of the density requires the repeated evaluation of both `EXP` and `ERFC`. In HiCu, `EXP` has been implemented using fourth order Chebychev interpolation, and `ERFC` with third order Chebychev interpolation. Both functions achieve an absolute accuracy of 10^{-12}, and yield a significant performance enhancement relative to full precision of the intrinsic functions provided by the compiler.

FIG. 4: Scaling of the RB3LYP(VWN3)/6-31G** exchange-correlation matrix build times for 4 target accuracies; $\tau_r = 10^{-4}, 10^{-5}, 10^{-6}, \text{and } 10^{-7}$.

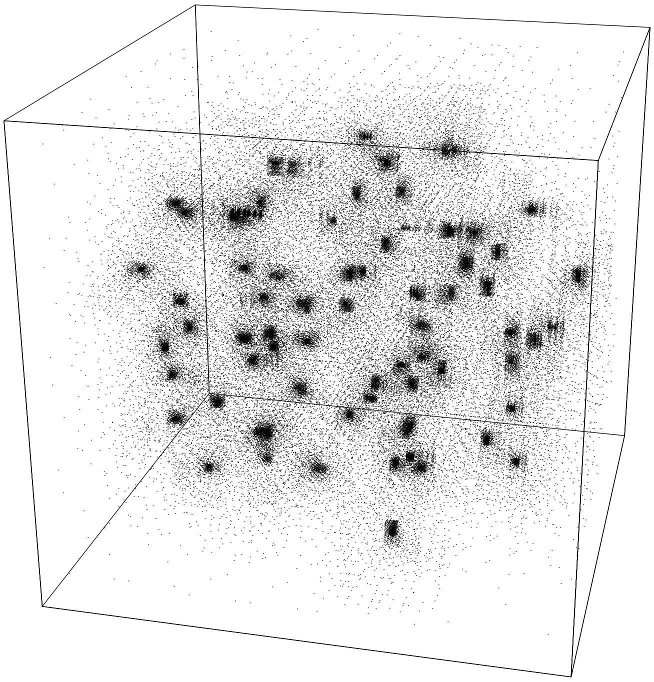


All programs have been compiled and run on a 550 MHz PIII running version 6.1 of RedHat LINUX[57], using version 3.03 of the Portland Group pgf90 compiler[58] with the options `-O2 -tp p6 -pc 64 -Mdalign -Mnoonetrip`.

IV. RESULTS AND DISCUSSION

Results have been obtained for the standard suite of three-dimensional water clusters ($N_{\mathrm{H_2O}} = 10, 20, 30, 40, 50, 60, \text{and } 70$), which have been used in a number of linear scaling studies[11,12,14,21,22,28,29,59–61]. Restricted B3LYP calculations using the VWN3 variation[62] and the 6-31G** basis set were carried out for this sequence of water clusters using $\tau_r = 10^{-4}, 10^{-5}, 10^{-6}, \text{and } 10^{-7}$, and are shown in Fig. 4. The guess density for these calculations was obtained by basis set switch[21] from a converged RHF/STO-3G density matrix. Timings corresponding to the mixed basis set have not been averaged in. After the switch, timings per $\mathbf{K}_{xc}^{\mathrm{DFT}}$ build were found to be nearly identical, and only values corresponding to the converged density are reported. That is, no averaging whatsoever has been performed. The results in Fig. 4 show a very early onset of linear scaling, which persists for decreasing values of τ_r.

FIG. 5: The HiCu grid for $(H_2O)_{70}$ and $\tau_r = 10^{-3}$. Also shown is the `BBox` of the CubeTree root node. The multi-scale structure of the grid results from recursive bisection rather than use of nuclear weights and spherical polar coordinate transformations.

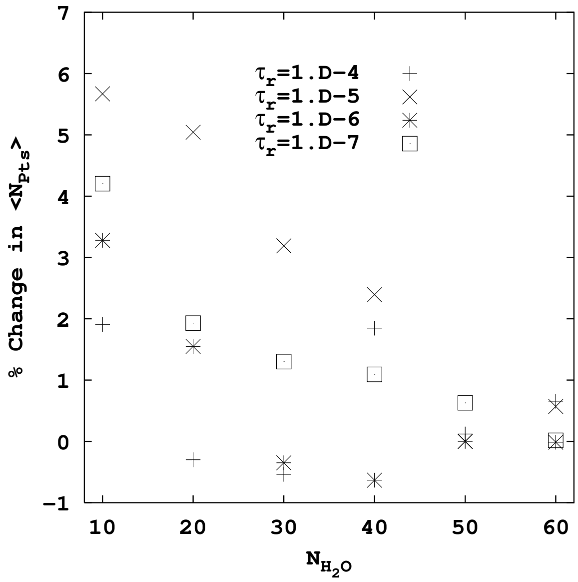


In Fig. 5, the HiCu grid obtained with $\tau_r = 10^{-3}$ is shown for $(H_2O)_{70}$. The grid points are contained by the CubeTree root node's `BBox`, and form a telescoping structure about each water molecule due to peaks in the electron density and multi-scale properties of the grid. The non-overlapping Cartesian properties of the HiCu grid should make fine grained domain decomposition in parallel implementations straight forward.

Unfortunately, direct comparison with the "industry standard" LinXC[63] is not possible due to license restrictions[64], and because of unpublished errors and undisclosed methods for achieving averaged CPU times. However, a qualitative comparison of performance may be achieved by comparison with the results of Ref. 63, wherein linear scaling of $\mathbf{J}$ and $\mathbf{K}_{xc}^{\mathrm{DFT}}$ is *not achieved* with basis sets and molecular systems nearly identical to the benchmark suites used in this study, even for clusters as large as $(H_2O)_{392}$. Here, the onset of linear scaling appears between $(H_2O)_{20}$ and $(H_2O)_{30}$.

A more direct comparison with other methods is possible on the basis of grid statistics. For the water clusters studied here, $\langle N_{\mathrm{Pts}}\rangle$ depends on the target accuracy, and approaches an asymptote as shown in Fig. 6. This may be due to surface-volume effects, as the domain of integration is approximately cubic, while the water clusters are approximately spherical. Thus, as the cluster size increases the interstitial region (cube-sphere) becomes small relative to the volume of the sphere. It is interesting that this effect becomes negligible ($\sim 2\%$) around $(H_2O)_{30}$, which likewise marks the onset of linear scaling.

FIG. 6: Variation in $\langle N_{\mathrm{Pts}}\rangle$ with increasing system size, N_{H_2O}, computed relative to the minimum number of points achieved, which occurs for $(H_2O)_{70}$.



In Fig. 7, the dependence of $\langle N_{\mathrm{Pts}}\rangle$ on ϵ_r is plotted, from which the efficiency curve

$$\epsilon_r = C\, 10^{-n\langle N_{\mathrm{Pts}}\rangle} \ , \tag{11}$$

is obtained, allowing a semi-quantitative comparison with other methods that have published statistics. In Table I, the values n and C are listed for the HiCu grid and for grids using the nuclear weights of Becke, Delley, and Lin, Jaffe and Hess (LJH)[38]. However, the HiCu values have been acquired using computed worst case errors while the values taken from LJH correspond to *target* accuracies. Also, different molecular systems and basis sets were used to arrive at these values. It should be noted that HiCu adapts to the basis set, with the size of the HiCu grid depending on the ratio of largest to smallest Gaussian exponent. When this ratio is large, the CubeTree will have to recur to higher resolution to achieve a given accuracy. Nevertheless, on the basis of this comparison, the HiCu grid appears competitive with nuclear weight schemes for low to medium accuracies.

A challenge facing all numerical DFT methods is to minimize the magnitude of grid errors, which may lead to discontinuities in the molecular potential energy surface. The HiCu grid error decreases systematically with decreasing values of τ_r, as shown in Fig. 8 for the RPBE0/6-31G** Ne_2 binding curve (See Ref.4 for details of this model chemistry). For $\tau_r = 10^{-6}$, the relative error in the total energy is on the order of 10^{-6}, with an absolute error in the binding curve of 10^{-3} hartree. Relative errors in forces should be of the same order as those in

FIG. 7: Scaling of the minimum $\langle N_{\rm Pts} \rangle$ with the maximum (worst) ϵ_r, taken over the entire series of water clusters. A fit to the last three points is also shown. Using all 4 points yields $\langle N_{\rm Pts} \rangle = 40\,\epsilon_r^{-.37}$, but a less agreeable fit.

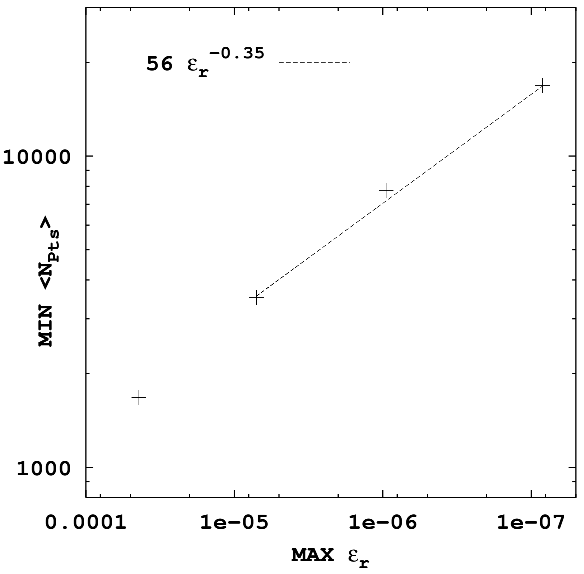


TABLE I: Parameters of the efficiency curve, Eq. (11), for various nuclear weight grids (taken from Lin, Jaffee, and Hess[38]). Also shown are the HiCu parameters derived from Fig. 7.

Grid	C	n
LJH weights	$10^{7.0}$	3.8
Delley weights	$10^{3.9}$	2.3
Becke weights	$10^{5.6}$	2.6
HiCu	$10^{5.0}$	2.8

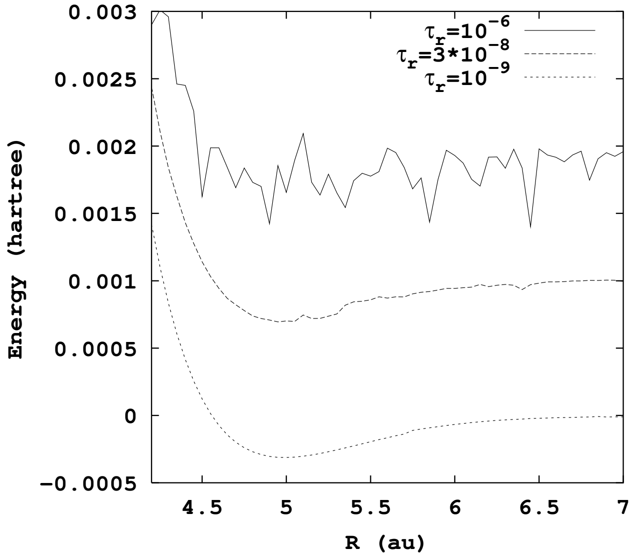
the total energy, and for *ab initio* molecular dynamics an accuracy of 5-6 digits should be sufficient for many applications. At $\tau_r = 3 * 10^{-8}$, errors in the binding curve have decreased by at least an order, and for $\tau_r = 10^{-9}$ a completely smooth curve obtains.

V. CONCLUSIONS

A new method, Hierarchical Cubature (HiCu), has been developed for achieving linear scaling computation of the exchange-correlation matrix within the Cartesian-Gaussian LCAO framework. HiCu has demonstrated an early onset of linear scaling for three-dimensional systems, which is persistent with increasing accuracy and non-trivial basis sets.

HiCu is unique in employing an entirely Cartesian grid for computation of the exchange-correlation matrix, overcoming nuclear cusps in the electron density with

FIG. 8: Binding curves of Ne_2 at the PBE0/6-31G** level of theory, with $\tau_r = 10^{-6}$ (shifted up by $2 * 10^{-3}$), $\tau_r = 3 * 10^{-8}$ (shifted up by 10^{-3}), and $\tau_r = 10^{-9}$.



a multi-scale approach rather than resorting to conventional spherical-polar coordinate transformations. The purely Cartesian representation supports the *k-d tree* data structure enabling efficient range queries, and provides an exact expression for the integration error at each level of resolution. This non-overlapping Cartesian representation may also enable the efficient implementation of fine grained parallelism.

One of the most demanding aspects of HiCu is the resolution of nuclear-electron cusps. As the Gaussian LCAO approximation tends toward completeness and the ratio of maximum to minimum Gaussian exponents becomes large, the HiCu grid can be expected to extend itself accordingly. However, as the nuclear-electron cusp is a weak singularity, the CubeTree will not become unbounded. Accordingly, the use of pseudopotentials should lead to substantial gains in efficiency. It may also be possible to employ coordinate transformations within each cube that yield more accurate results than the native rule. During the development of HiCu, a number of one dimensional transformations were tested for regularization of the integrand, including Pederson's variational mesh approach[40]. These transformations were found to yield significant advantages in the case of a one dimensional density, but to be ineffective in three dimensions. Three dimensional transformations such as those described in Refs.[65,66] may be more effective, but have not been tried.

A number of similarities exist between HiCu and the Quantum Chemical Tree Code QCTC for computation of the Coulomb matrix $\mathbf{J}$[11,28,29]. As the HiCu grid is traversed, work is performed only for overlapping nodes.

Likewise QCTC avoids work by invoking the multipole approximation for nodes with insignificant penetration. With a sufficiently accurate multipole approximation, the evaluation of penetration acceptability criteria (PAC) in the current implementation of QCTC dominates when representation of the density proceeds to the finest resolution. The more efficient *k-d tree* methods developed here for accessing overlapping elements of the density may thus be applied directly to the next generation of QCTC.

ACKNOWLEDGMENTS

This work has been carried out under the auspices of the US Department of Energy under contract W-7405-ENG-36. Thanks to Tommy Sewell and Jeff Hay for a careful reading of the manuscript.

REFERENCES

[†] Electronic address: `MChalla@LANL.Gov`; URL: `http://www.t12.lanl.gov/home/mchalla/`
[1] A. D. Becke, J. Chem. Phys. **98**(2), 1372 (1993).
[2] P. J. Stephens, F. J. Devlin, C. F. Chabalowski, and M. J. Frisch, J. Phys. Chem. **98**(45), 11623 (1994).
[3] C. W. Bauschlicher, Chem. Phys. Lett. **246**(1-2), 40 (1995).
[4] C. Adamo and V. Barone, J. Chem. Phys. **110**(13), 6158 (1999).
[5] C. Adamo, A. di Matteo, and V. Barone, Adv. Quant. Chem. **36**, 45 (2000).
[6] S. F. Boys, Proc. Roy. Soc. **A200**, 542 (1950).
[7] S. Huzinaga, Comp. Phys. Rep. **2**(6), 281 (1985).
[8] E. R. Davidson and D. Feller, Chem. Rev. **86**, 681 (1986).
[9] I. Shavitt, Israel J. Chem. **33**(4), 357 (1993).
[10] P. M. W. Gill, Adv. Q. Chem. **25**, 141 (1994).
[11] M. Challacombe and E. Schwegler, J. Chem. Phys. **106**, 5526 (1997).
[12] E. Schwelger, M. Challacombe, and M. Head-Gordon, J. Chem. Phys. **106**(23), 9708 (1997).
[13] E. Schwegler and M. Challacombe, Theo. Chem. Acc. (2000), to appear.
[14] E. Schwelger and M. Challacombe, J. Chem. Phys. **111**(14), 6223 (1999).
[15] J. Almlöf, K. Faegri, and K. Korsell, J. Comp. Chem. **3**(3), 385 (1982).
[16] M. Häser and R. Ahlrichs, J. Comp. Chem. **10**(1), 104 (1989).
[17] P. O. Löwdin, J. Chem. Phys **18**, 365 (1950).
[18] P. O. Löwdin, Adv. Phys. **5**, 3 (1956).
[19] C. C. J. Roothaan, Rev. Mod. Phys. **23**, 69 (1951).
[20] G. G. Hall, Proc. Roy. Soc. **208 A**, 328 (1951).
[21] M. Challacombe, J. Chem. Phys. **110**(5), 2332 (1999).
[22] M. Challacombe, Comp. Phys. Comm. (2000), to appear.
[23] J. M. Pérez-Jordá and W. T. Yang, Chem. Phys. Lett. **241**(4), 469 (1995).
[24] R. E. Stratmann, G. E. Scuseria, and M. J. Frisch, Chem. Phys. Let. **257**(3-4), 213 (1996).
[25] L. Lou, J. Comp. Phys. **157**(1), 404 (2000).
[26] J. A. Pople, P. M. W. Gill, and B. G. Johnson, Chem. Phys. Lett. **199**(6), 557 (1992).
[27] G. R. Ahmadi and J. Almlöf, Chem. Phys. Lett. **246**(4-5), 364 (1995).
[28] M. Challacombe, E. Schwegler, and J. Almlöf, *Computational Chemistry: Review of Current Trends* (World Scientific, Singapore, 1996), pp. 53–107.
[29] M. Challacombe, E. Schwegler, and J. Almlöf, J. Chem. Phys. **104**(11), 4685 (1996).
[30] A. D. Becke, J. Chem. Phys. **88**(4), 2547 (1998).
[31] C. W. Murray, N. C. Handy, and G. J. Laming, Mol. Phys. **78**(4), 997 (1993).
[32] J. M. Pérez-Jordá, A. D. Becke, and E. San-Fabián, J. Chem. Phys **100**(9), 6520 (1994).
[33] J. M. Pérez-Jordá, J. Chem. Phys. **101**(2), 1738 (1994).
[34] O. Treutler and R. Ahlrichs, J. Chem. Phys. **102**(1), 346 (1995).
[35] M. E. Mura and P. J. Knowles, J. Chem. Phys. **104**(24), 9848 (1996).
[36] V. Termath and J. Sauer, Chem. Phys. Lett. **255**(1-3), 187 (1996).
[37] M. Krack and A. M. Koster, J. Chem. Phys. **108**(8), 3226 (1998).
[38] Z. J. Lin, J. E. Jaffe, and A. C. Hess, J. Phys. Chem. A **103**(13), 2117 (1999).
[39] P. M. Boerrigter, G. Te Velde, and E. J. Baerends, Int. J. Quant. Chem. **33**(2), 87 (1988).
[40] M. R. Pederson and K. A. Jackson, Phys. Rev. B. **41**(11), 7453 (1990).
[41] M. R. Pederson, D. V. Porezag, J. Kortus, and D. C. Patton, Phys. Stat. Sol. B **217**(1), 197 (2000).
[42] T. Kato, Commun. Pure Appl. Math **10**, 151 (1957).
[43] J. L. Bentley and J. H. Friedman, Comp. Surv. **11**(4), 397 (1979).
[44] J. L. Bentley, Comm. ACM **23**(4), 214 (1980).
[45] V. Gaede and O. Günther, ACM Comp. Surv. **30**(2), 170 (1998).
[46] A. H. Stroud, *Approximate Calculation of Multiple Integrals*, Prentice-Hall series in automatic computation (Prentice-Hall, New Jerzey, 1971).
[47] D. A. Dunavant, Int. J. Num. Meth. Eng. **23**(3), 397 (1986).
[48] M. Beckers and A. Haegemans, J. Comp. App. Math. **35**(1-3), 109 (1991).
[49] R. Cools and P. Rabinowitz, J. Comp. App. Math. **48**(3), 309 (1993).
[50] P. van Dooren and L. de Ridder, J. Comp. App. Math. **2**(3), 207 (1976).
[51] A. C. Genz and A. A. Malik, J. Comp. App. Math **6**(4), 295 (1980).

[52] J. Berntsen, T. O. Espelid, and A. Genz, ACM Trans. Math. Soft. **17**(4), 437 (1991).

[53] J. Berntsen and T. O. Espelid, ACM Trans. Math. Soft. **17**(2), 233 (1991).

[54] A. C. Genz and A. A. Malik, SIAM J. Num. Anal. **20**(3), 580 (1983).

[55] R. Cools and A. Haegemans, Numer. Math **55**, 735 (1989).

[56] M. Challacombe and E. Schwegler, MondoSCF *v0.2, A program suite for massively parallel, linear scaling SCF theory*, http://www.t12.lanl.gov/home/mchalla/ (2000).

[57] Red Hat, *Red Hat Intel Linux 6.1*, http://www.redhat.com (2000).

[58] The Portland Group, *pgf90 v3.0.3*, http://www.pgroup.com/ (2000).

[59] J. C. Burant, R. E. Stratmann, and M. J. Frisch, J. Chem. Phys. **105**(19), 8969 (1996).

[60] E. Schwegler, *Linear Scaling Computation of the Hartree-Fock Exchange Matrix*, Ph.D. thesis, University of Minnessota (1998), at http://www.t12.lanl.gov/home/mchalla/.

[61] C. Ochsenfeld, C. A. White, and M. Head-Gordon, J. Chem. Phys. **109**(5), 1663 (1998).

[62] R. H. Hertwig and W. Koch, Chem. Phys. Lett. **268**(5-6), 345 (1997).

[63] G. Scuseria, J. Phys. Chem. **103**(25), 4782 (1999).

[64] E. K. Wilson, Chem. Eng. News **77**(28), 27 (1999).

[65] A. Genz and R. E. Kass, J. Comp. Graph. Stat. **6**(1), 92 (1997).

[66] V. Sladek, J. Sladek, and M. Tanaka, Eng. Anal. Bound. Elem. **20** (1997).