ELSEVIER

3 May 1996

Chemical Physics Letters 253 (1996) 268–278

CHEMICAL PHYSICS LETTERS

Linear scaling density functional calculations via the continuous fast multipole method

Christopher A. White [a], Benny G. Johnson [b], Peter M.W. Gill [c], Martin Head-Gordon [a,*]

[a] *Department of Chemistry, University of California at Berkeley, Berkeley, CA 94720, USA*
[b] *Q-Chem, Inc., 317 Whipple St., Pittsburgh, PA 15218, USA*
[c] *Department of Chemistry, Massey University, Palmerston North, New Zealand*

Received 8 November 1995; in final form 23 January 1996

Abstract

We apply the linear scaling continuous fast multipole method (CFMM) to form the J matrix for molecular density functional calculations. Our implementation involves a new definition of charge distribution extent that bounds absolute errors. We efficiently treat short range interactions via a J matrix engine without fully uncontracting the basis. Calculations on 1-d, 2-d and 3-d carbon systems with the 3-21G basis establish crossover points versus the conventional approach, and yield linear scaling coefficients between 10^6 and 10^8 floating point operations, depending on dimensionality. The CFMM plus J engine is a dramatic improvement for molecules over 50 to 100 atoms.

1. Introduction

Over the last several years, a variety of methods have been introduced addressing the scaling issues for calculations of the Coulomb potential in systems composed of classical point charges [1–6] [1]. Interest in reducing the scaling of the Coulomb problem has now shifted to ab initio electronic structure calculations [7–10] [2], particularly for self-consistent field theories including Kohn–Sham density functional theory (DFT) and Hartree–Fock calculations where it is rate-determining at present. To address this need, we have introduced the continuous fast multipole method (CFMM) [7]. This method, a generalization of Greengard and Rokhlin's fast multipole method (FMM) [1–5], allows one to calculate the Coulomb interactions of a collection of finite extent distributions represented by continuous functions, in work scaling linearly with system size.

* National Science Foundation Young Investigator 1993–8, Alfred P. Sloan Fellow 1995–1997, and David and Lucille Packard Fellow, 1995–2000, and author to whom correspondence should be addressed.

[1] Ref. [2] gives a recent general audience review of fast-summation algorithms.

[2] The authors of Ref. [10] have implemented a version of the CFMM that they choose to call the Gaussian FMM (GFMM), and an extension they term the Gaussian very fast multipole method (GvFMM).

PII S0009-2614(96)00175-3

The Coulomb interaction matrix, or J matrix, of DFT is conventionally obtained by contracting the two-electron repulsion integrals (ERIs) with the one-particle density matrix. This operation constitutes the rate-determining step in current 4-center molecular DFT calculations as a result of the creation of a large number of intermediate quantities (ERIs) to form a small number of final quantities of interest (the J matrix). With the use of integral prescreening [11–14], J matrix construction scales as $O(N^2)$, where N is the dimension of the one-particle basis. Diagonalization of the effective Hamiltonian is an $O(N^3)$ step, but with such a small coefficient that it is dominated by the Coulomb problem for the largest calculations yet performed this way (several hundred atoms).

Very recently we have introduced a procedure for forming the *exact* J matrix *without* explicit assembly of the ERIs, by pre-summing the density matrix into the underlying recurrence relations [15]. This J matrix engine, and related methods [16,17], are probably the best exact N^2 methods available at present, although faster N^2 methods can be obtained by the use of approximate auxiliary basis expansions of the density [18]. This Letter reports the dramatic speedups that can be obtained by efficiently applying the linear scaling CFMM to calculate the J matrix, while still retaining a bounded error that is comparable to the precision obtained by explicit ERI formation.

The CFMM divides Coulombic interactions into local (near-field) and distant (far-field) contributions to the J matrix. A linear scaling method based on the FMM rapidly handles the distant contributions by collectivizing charge distributions into multipoles. The remaining local contributions must be evaluated explicitly, but their number is only linear, scaling as $O(NM)$ where M is an effective number of neighboring charges, and is independent of the system size. Here, we apply our J matrix engine to handle the near-field contributions in a rigorous yet still highly efficient manner.

In the remainder of this Letter, we will first discuss theoretical issues related to applying the CFMM to treat molecular charge distributions in Section 2. Section 3 briefly describes several key points concerning the practical implementation of a CFMM method for the J matrix problem. Numerical tests of the accuracy and speed of our current implementation of the CFMM for density functional calculations are presented in Section 4. We consider model systems of both 1-, 2- and 3-dimensional character, and show that the linear scaling CFMM revolutionizes the speed of J matrix assembly for molecules in the range of 50 to 100 atoms and upwards.

2. Theoretical issues

2.1. An extent definition for molecular charge distributions

Linear scaling J matrix assembly involves two requirements. First, the number of charge distributions must grow linearly with the size of the system. Second, a linear scaling method such as the CFMM must calculate the Coulomb interactions of this $O(N)$ collection of charge distributions. In molecular calculations the charge distributions arise from products of atom centered shells, known as shellpairs. Since the number of shells grows linearly with the size of the system, the number of shellpairs formally grows quadratically. For small molecules, one cannot avoid this scaling, but as the size of the system increases, shellpair-based integral pre-screening techniques [11–13] ensure linear growth in the number of significant charge distributions.

Closely related to these pre-screening techniques, the CFMM achieves linear scaling by determining the point at which two distributions begin to Coulombically interact as point charges (or higher point multipoles). This leads to the definition of an extent which represents the closest approach of two interacting point distributions. For a pair of Gaussian charge distributions, this extent [3] takes an exceptionally simple form,

$$r_{\mathrm{ext}} = \frac{1}{2}\sqrt{\frac{2}{p}\ln(\epsilon)}\,, \tag{1}$$

[3] This definition takes a slightly different mathematical form than given in Ref. [6]. This form is the first order correction obtained through a Taylor expansion of erf(x).

where p is the Gaussian exponent. This version of the extent ensures the deviation of the classical approximation from the exact result will be a factor of ϵ smaller than the absolute magnitude of the interaction.

In molecular systems, charge distributions appear as products of Gaussian basis functions. These charge distributions always contain a Gaussian pre-factor which arises from the Gaussian product rule and depends directly upon the distance AB between the centers of the original shells. If one relaxes the error tolerance from the relative precision of the integral, to an absolute tolerance as compared to the other J matrix contributions, one can include the Gaussian pre-factor in the definition of extent. In this case we have

$$r_{\text{ext}} = \frac{1}{2}\sqrt{\frac{2}{p}\ln(\epsilon) - AB^2}\,. \tag{2}$$

Eq. (2) no longer ensures that the error is small in comparison to the magnitude of the interaction; it simply ensures the error in the contribution to the J matrix is smaller than some ϵ. As would be expected, this definition leads to much smaller extents with negligible increase in the errors introduced in final the J matrix.

A further modification of the extent definition could be made by including screening based on the density. Density-based cutoffs are particularly useful in forming an increment to the J matrix based on a difference density. We do not exploit this technique here, but it can be applied to reduce the number of charge distributions to interact, as well as to slightly modify the extent definition, and therefore the effective number of neighbors with which a given distribution explicitly interacts. These advantages will come at the price of more rapid and unpredictable error accumulation, and will not be explored here.

2.2. *Treatment of far-field interactions in the FMM and CFMM*

Using the extent definition given by Eq. (2), we can effectively partition the Coulomb interactions into near-field and far-field sets. The near-field interactions are evaluated using standard integral techniques, which in our case is the J matrix engine [15]. The far-field interactions are treated using the CFMM-based method. For point charge systems, the FMM itself contains several fundamental constraints which can limit the number of these interactions which can be evaluated using multipoles. The introduction of extent in the CFMM introduces a further constraint, whose effect on the near-field versus far-field partitioning must also be considered.

The fraction of interactions performed directly without the use of multipole expansions (D) for a point charge system approaches the ratio of number of boxes, N_{ws}, with which charges in a reference box interact directly, to the total number of boxes into which the system is partitioned, N_{box}

$$D = \frac{N_{\text{ws}}}{N_{\text{box}}}. \tag{3}$$

The well-separatedness (WS) index is defined such that WS = 1 indicates direct interactions end at nearest neighbor boxes, while WS = 2 indicates that up to next-nearest neighbor boxes are directly evaluated and so forth. The connection between WS and N_{ws} depends on the dimensionality, d, of the system via

$$N_{\text{ws}} = (2\text{WS} + 1)^d, \tag{4}$$

neglecting edge effects. The total number of boxes is also a power of the dimensionality multiplied by the tree depth, t

$$N_{\text{box}} = 2^{d(t-1)}; \tag{5}$$

where t is defined as one plus the number of binary subdivisions of the parent box (which encloses the system) necessary to reach the finest level boxes.

From Eq. (3), the fraction of direct interactions is independent of the number of particles in the system assuming a homogeneous distribution with each box containing at least one particle, and neglecting edge effects. For a fixed accuracy (as specified by the WS value, which is usually 2) the fraction of interactions done directly can be reduced by simply increasing the tree depth, as is clear from Eq. (5). Furthermore, substitution of Eqs. (4) and (5) into Eq. (3) shows that the fraction of direct interactions for a given WS value and tree depth is a constant raised to the dimensionality of the system. For instance, to ensure no more than a specified fraction of direct interactions, much deeper trees are required in 1-d systems, than for 2-d systems.

The CFMM takes account of the extent of distributions by assigning a WS value to each distribution based on the following equation:

$$WS = \max(2\lceil r_{ext}/l \rceil, \mathrm{WS}_{ref}), \quad (6)$$

where l is the length of a box, and WS_{ref} is the WS value chosen for point charges. Thus the WS value (and hence the number of interacting neighbor boxes) is determined by the extent of the distributions. For distributions covering the entire system, N_{ws} effectively becomes N_{box}; such distributions must interact with all other distributions directly.

The useful limit on tree depth in the FMM is set by the requirement that each box should be populated. By contrast, the dynamic definition of WS value for the continuous distributions treated in the CFMM means that the point of diminishing returns (with respect to decreasing the fraction of direct interactions) can be reached at smaller tree depths. As the depth of the CFMM tree increases, l decreases while the spatial extent, r_{ext}, of the distributions remains constant. Therefore, from Eq. (6), the WS value of a given distribution potentially increases, and increasing the depth of the CFMM tree does not necessarily decrease the fraction of direct interactions.

To achieve linear scaling using the CFMM, both the density of the system and the extent of the charge distributions must remain constant as one increases the size of the system. This ensures that at some point the depth of the CFMM tree may increase without increasing the WS values for the distributions. Fortunately this is indeed the situation which pertains to molecular calculations, where the extent of the shellpairs is a function only of the atom types, and the densities are approximately invariant to size.

3. Implementation issues

3.1. Contracted shellpairs

A given contracted shellpair is a linear combination of primitive shellpairs, consisting of different pairs of exponents. By the Gaussian product theorem, primitives belonging to a single contracted shellpair are potentially centered in different boxes. In this case, one must split the original contracted shellpair into (at least) two derived shellpairs, each centered in a different box. This presents no significant problem for the far-field CFMM. However, to exploit the speed and design of conventional techniques for the evaluation of the near-field interactions, it is desirable to keep the shellpairs as highly contracted as possible. Therefore we retain fully

Table 1
C_{150} timing breakdowns (in seconds, recorded on a DEC Alpha 3000 Model 600 workstation) for CFMM calculations on 1-d, 2-d and 3-d systems, as a function of the depth of the CFMM tree

System	Tree depth	Near-field		Far-field		Total	
		via ERI's	J engine	10 poles	25 poles	10 poles	25 poles
1d	direct	2413	597	–	–	–	–
	4	1455	375	13	73	388	448
	5	715	189	13	76	202	265
	6	355	96	18	86	114	182
	7	181	53	55	135	108	188
2d	direct	26629	6229	–	–	–	–
	4	13694	3227	38	311	3265	3538
	5	6019	1215	101	1993	1316	3208
	6	3346	786	1127	19731	1913	20517
3d	direct	116722	25631	–	–	–	–
	4	72070	13000	241	5579	13241	18579
	5	33851	5882	6032	–	11914	–

Direct calculations are included for comparison. Calculations were performed using both 10 and 25 poles, and the 3-21G basis. Near field times via both J engine and conventional ERI methods are presented, to show the value of a J engine-based method; totals are based on the J engine plus CFMM combination. The diagonalization time is approximately 900 s for comparison.

contracted shellpairs unless required to split them because of primitive centers being located in multiple boxes. This is more efficient than the possible alternative of completely uncontracting the basis. In the latter case, the efficiency of the CFMM might compare well against conventional methods using the same *uncontracted* basis, but would compare more poorly against conventional methods using the original *contracted* basis.

3.2. Data locality and the J matrix engine

Elsewhere we have described a new direct method for forming a J matrix without explicit evaluation of ERIs [15]. This J matrix engine has been shown to yield significant speedups relative to methods based on ERI evaluation, with identical precision. Here we focus on the importance of the J engine as it relates to the CFMM. Table 1 shows several pieces of timing information for our 1-d, 2-d, and 3-d model systems containing 150 carbons. For each depth of CFMM tree, the time for the calculation of the near-field interactions is listed as obtained using the J engine and a conventional HGP-PRISM [19] algorithm. The main contribution of the J engine can be seen in the ratio of the near-field to far-field timings using these two methods. The use of the J engine drastically reduces the amount of near-field time for these calculations making it much easier to obtain equal balances of far-field and near-field timings.

3.3. Memory constraints and adaptive boxing

As was discussed in Section 2.2 the depth of the CFMM tree relates directly to the total number of possible interactions performed using multipoles. The memory requirements for applying the CFMM to a 3-d homogeneous system grows linearly with the size of the system. However for lower dimensionalities, this is not necessarily true, because substantial numbers of boxes may contain no charges, with the extreme example being a linear system. To retain the linear growth in memory demand with system size, our implementation ensures that memory is not allocated for the empty boxes, so that these vacant areas of space are simply ignored. Memory requirements are thus determined only by the number of boxes enclosing charge, not by the depth of the CFMM tree. This allows our code to reach depths of 6 and 7 tiers for linear chains with the use of less than 128 MB of memory.

4. Results and discussion

We have implemented our CFMM based method within the Q-Chem computational chemistry package [20]. Our current version supports s, p and sp shells with work continuing to extend this to arbitrary angular momenta. The CFMM far-field implementation is linked to an arbitrary angular momentum J engine for efficient treatment of the near-field contributions. This combination allows us to assess the comparative performance of the CFMM relative to conventional quadratic methods by timings on illustrative model systems. All reported calculations were performed on a DEC Alphastation 3000 Model 600.

4.1. Model systems

Successful application of linear scaling methods such as the CFMM to electronic structure calculations is possible only for large molecules, where in this context the meaning of the oft-used adjective 'large' does not relate only to the total number of constituent atoms. Instead it is closer to the ratio of the number of atoms to an effective coordination number of a single atom, and thus depends directly on both numbers of atoms and the dimensionality of the system. A system growing mainly in a single dimension such as a linear chain becomes *large* much more rapidly than a 3-d diamond structure; a 10-atom linear chain separates the carbons at the ends to a much larger degree then a 10-atom globular cluster separates the atoms comprising its bounding surface.

We take as our model systems, the 1-d, 2-d, and 3-d elemental forms of carbon. The 1-d system consists of a linear chain of carbons having a bond length representative of polyacetylene. The 2-d system consists of a graphitic structure having the equilibrium bond length of graphite. Finally, the 3-d system consists of a diamond structure having the equilibrium bond length of diamond. These 3-d systems are in some ways worst-case examples, since they are far denser and more compact than most

molecules of chemical interest. The contrast in the behaviors of the different dimensionality systems will allow us to roughly characterize the range of performance expected for applying the CFMM to molecular systems.

We report timings for four distinct types of methods, using the above systems with the standard 3-21G split valence basis. The first procedure is J matrix assembly via explicit ERI evaluation (ERI's are evaluated by the HGP-PRISM method [19]), which yields asymptotic quadratic scaling. The second method is our recently introduced J matrix engine [15], which avoids explicit ERI evaluation, but is also asymptotically quadratic. These calculations in the most efficient manner possible, without any uncontracting or splitting of the shellpairs. The results are typical of the best that can currently be achieved in a quadratic approach, without approximations such as auxiliary basis expansions.

We compare these quadratic methods against the linear scaling CFMM with $\epsilon = 10^{-10}$ (similar to the precision sought in ERI evaluation), and $WS_{ref} = 2$, using the J matrix engine for near-field contributions. These CFMM calculations are performed with two choices of multipole truncation, 10 poles and 25 poles. These choices are consistent with an accuracy in far-field contributions to the J matrix of about 10^{-7} and 10^{-10} respectively. More careful optimization of the choice of multipole cutoff, WS value, and ϵ to achieve a given target accuracy is of course possible: our purpose here is to assess the speed of the CFMM with parameters that are possibly more conservative than optimal.

4.1. Linear scaling in the CFMM

The timing results for calculations on the 1-, 2- and 3-dimensional model systems are summarized in Figs. 1, 2 and 3. The CFMM curves (including near-field J engine contributions) show great similarity to the FMM behavior we have previously documented [3], particularly in Fig. 1 for the 1-d

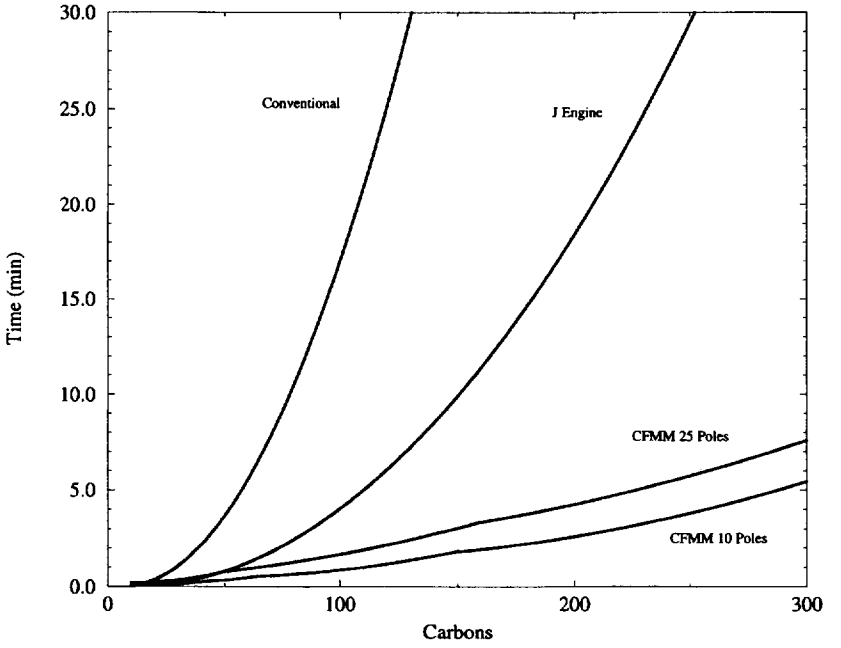

Fig. 1. Timings for one-dimensional carbon chains. This graph shows the time to assemble the J matrix for a series of one-dimensional molecules. The 1-d system consists of a linear chain of carbons having a bond length representative of polyacetylene using the standard 3-21G split valence basis. Conventional corresponds to explicit J matrix formation using the HGP-PRISM integral method. J engine corresponds to explicit J matrix formation using only the J engine method. 10 Poles and 25 Poles correspond to CFMM calculations using 10 and 25 poles, respectively, for the far-field contributions to the J matrix and the J engine method for the near-field. The crossover with our best quadratic technique (the J engine) occurs in the range of 20–30 carbon atoms. As discussed in the text, the coefficient of linear scaling (obtained by connecting peaks or valleys) is approximately 10^6 FLOPs per basis function.

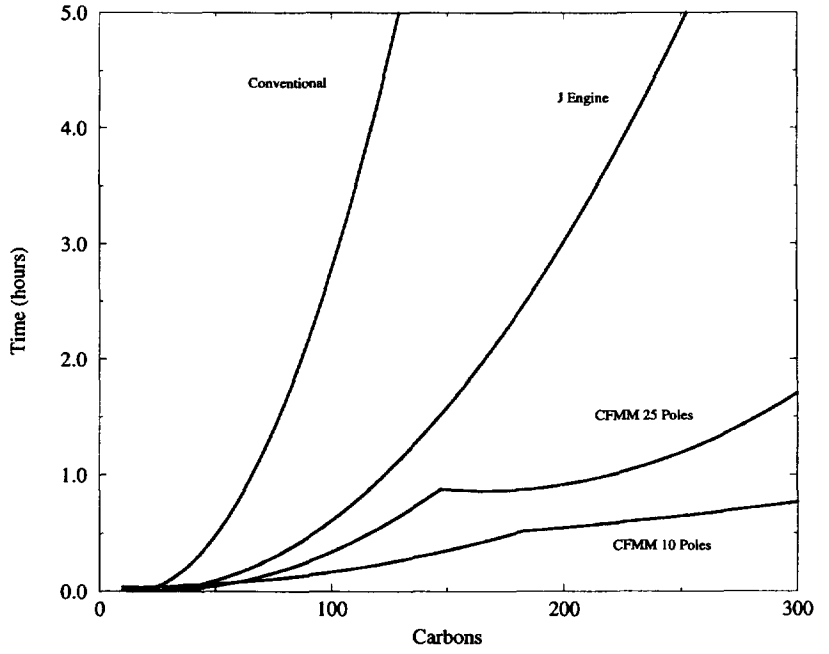

Fig. 2. Timings for two-dimensional carbon sheets, where the meaning of the four curves is as described in Fig. 1. The 2-d system consists of a graphitic structure having the equilibrium bond length of graphite using the standard 3-21G split valence basis. The crossover with our best quadratic technique (the J engine) occurs in the range of 30–50 carbon atoms. The coefficient of linear scaling (obtained by connecting peaks or valleys) is approximately 10^7 FLOPs per basis function (see text for discussion).

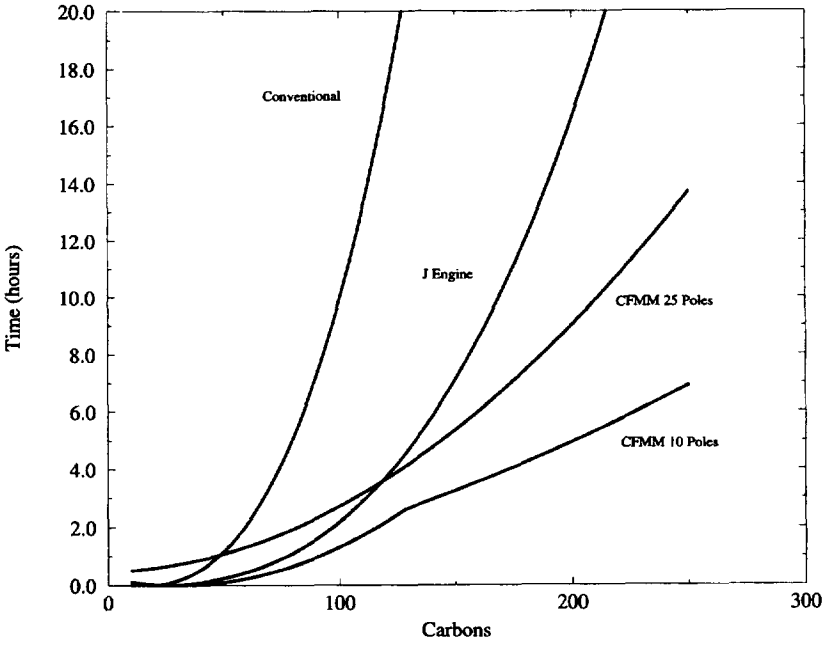

Fig. 3. Timings for three-dimensional carbon clusters, where the meaning of the four curves is as described in Fig. 1. The 3-d system consist of a diamond structure having the equilibrium bond length of diamond using the standard 3-21G split valence basis. The crossover with our best quadratic technique (the J engine) occurs in the range of 50–100 carbon atoms.

systems. Increasing depths of FMM tree cause movement between locally quadratic curves (for fixed tree depth, the CFMM scales quadratically [3,7]). Linear scaling is achieved in the sense that straight line tangents can be drawn bounding the CFMM curves from above (through the cusps connecting different numbers of tiers), or below. This manifestly establishes linear scaling in the 1-d case. In the 2-d and 3-d cases, we do not proceed to such deep trees (five in 3-d, six in 2-d and seven in 1-d are the highest levels employed in these calculations) and therefore the evidence for linear scaling in Figs. 2 and 3 is less strong, with the most extreme case being the 25 pole curve of Fig. 3, where only the quadratic curve for a five-level tree is evident (the cross-over to six tiers occurs for larger systems than those considered).

The most important results that can be drawn from Figs. 1–3 are the positions of the crossovers relative to conventional quadratic methods. For all three dimensionalities we obtain crossovers at system sizes which are not beyond the realm of applicability of the conventional approaches: in all three cases the crossover relative to ERI evaluation occurs below 100 atoms. Subject to attaining sufficient accuracy, which we discuss in the following subsection, this clearly establishes the value of the CFMM for accelerating the evaluation of the Coulomb contribution to the J matrix in DFT. As is clear from considering the large molecule region to the right of Figs. 1 to 3, this speedup is absolutely dramatic by the time systems of several hundred atoms are considered: well over an order of magnitude.

While it is immediately evident from comparing Eqs. (1) and (2) that the new definition of extent advocated here reduces the effective WS value assigned to distributions, it is interesting to assess its effectiveness in these calculations. As an example, we have repeated the 2-d calculation on C_{150} reported in Table 1 using the original definition of extent, and find that the calculation runs 4 times slower using 10 poles, and 2 times slower using 25 poles, due to extra near-field work.

As discussed in Section 2.2, the treatment of extent in the CFMM can lead to new effects in the size-dependence of computational effort. This is manifested in the data plotted in Figs. 2 and 3, where there is extra structure corresponding to system sizes at which distributions become point charge-like for a particular tree depth. Since we are dealing with carbon systems only, the distribution of Gaussian exponents is rather small, and a large number of interactions (e.g. between distributions in boxes separated by two intervening boxes) can become 'classical' all at once. This explains some otherwise strange effects such as molecules having more atoms running faster than those having less atoms.

The timings for the CFMM-based calculations depend quite strongly upon the level of multipoles chosen, as is evident in the differences between results with 10 poles and 25 poles in Figs. 1–3. The difference in timing results primarily from the $O(L^4)$ scaling of the translation operators of CFMM, and indicates that methods which improve the efficiency of far-field evaluation are desirable. The fact that far-field contributions are significant is a reflection of the efficient treatment of the short range interactions via the J matrix engine. From Table 1, if we were employing conventional ERI evaluation for this step, the near-field contributions would increase by a factor of roughly four in these calculations. Finally, in the context of the timing breakdowns of Table 1, it is interesting to consider the relative time taken to diagonalize the Fock matrix, which is approximately 900 s for the C_{150} case. Comparing with Table 1, diagonalization is already dominating J matrix formation in 1 dimension, and for 10 poles it is roughly comparable to J matrix formation in 2-d, while for 3-d C_{150}, it remains insignificant.

One final feature of considerable interest which can be roughly extracted from Figs. 1–3 is an estimate of the effective coefficient of linear scaling for these model systems. This gives a coefficient which is specific to the computers employed in the calculations, but with an estimate of the floating point operations (FLOPs) per second performance of the code, we can convert this data to a transferable form in units of FLOPs per basis function. We estimate approximately 10^6 FLOPs per basis function in 1-d, and roughly 10^7 FLOPs per basis function in 2-d, while the diamond-like structures appear to correspond to roughly 10^8 FLOPs per basis function, although we do not have enough data for a reliable estimate.

These coefficients very roughly summarize the current performance of the CFMM; of course further improvements will occur in the future, and we note

that the coefficients may be larger for more extended basis sets and elements in lower rows of the periodic table. They are already enough to indicate that if diagonalization of the Hamiltonian proceeds with a coefficient of between 10 and 100 N^3, then we can expect this step to be rate determining on calculations above roughly 1000 to 3000 basis functions, depending on dimensionality, consistent with our C_{150} results discussed above.

4.2. CFMM errors

The RMS errors in J matrix elements (considering only elements which are above our elimination threshold of 10^{-10}) for our 1-d model systems are portrayed in Fig. 4. This figure includes results obtained by performing CFMM calculations using several depths of CFMM tree for both 10 and 25 multipoles. The dotted lines connecting tree depths indicated the points at which one must increase the tree depth in order to obtain linear scaling. Thus the errors involved in an actual calculation would be given by the path traced by the dotted lines. The calculations performed using 25 poles show relatively little change with the size of the molecule indicating we are effectively obtaining the far-field contribution to the J matrix to the same level of accuracy (10^{-10}) that we evaluate ERIs in the quadratic algorithm. This illustrates the ability of the CFMM to give large speedups with little or no sacrifice in accuracy. In contrast, the 10 pole calculations show a gradual increase in accuracy as we increase the amount of near-field work thus indicating a poorer description of the far-field interactions, as would be expected. These results have no strong dependence on the dimensionality of the system.

The errors within the CFMM procedure arise from two distinct sources. The simplest source is the level of multipoles used to perform the calculations. This error is essentially given by standard FMM

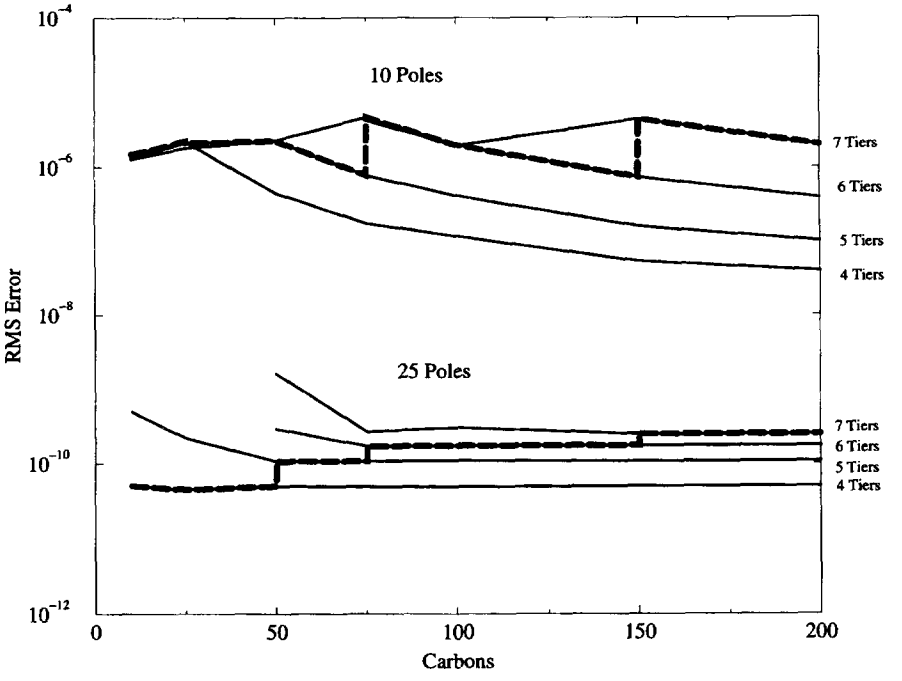

Fig. 4. RMS errors per J matrix element for one dimensional carbon chains. This graph shows the RMS absolute error in a J matrix element for our 1-d model systems. The two collections of horizontal lines correspond to the use of two different levels of multipoles (10 and 25) for the treatment of far-field interactions. In all calculations the extent definition given by Eq. (2) was used with $\epsilon = 10^{-10}$. Within each set of lines, we present data for several depths of CFMM trees, the dotted line dictates when the choice to increase the tree depth is made for these systems and thus should be considered the actual errors for a real calculation.

error estimates with the appropriate WS definition. This represents the difference between the 10 and 25 pole calculations shown in Fig. 4. The second source of error is the determination of extent. As described in Section 2, we use an extent criterion containing a single adjustable threshold designed to achieve a given relative precision (here set to $\epsilon = 10^{-10}$). This contribution dominates the errors seen for the 25 pole calculations, since this level of multipoles gives essentially machine precision for the far-field interactions.

As molecular size increases, numerical roundoff errors become important. In the case of conventional techniques, this problem is driven by the numbers of intermediate quantities (the two-electron integrals) used to produce the J matrix growing quadratically with molecular size. In the case of the CFMM, the number of intermediate quantities grows only linearly with size, and the error in all contributions remains bounded. For this reason error accumulation in the CFMM can actually be less severe than in quadratic methods. For this reason, one could argue that the CFMM is in fact more accurate then conventional methods in the limit of really large systems.

5. Conclusions

In this paper we have reported details of the application of the CFMM to solve the Coulomb problem in density functional calculations. This first involved technical advances such as efficient treatment of short-range interactions, minimal uncontracting of the shellpairs, and most importantly an improved treatment of the extent of shellpair distributions. The second issue addressed is the characterization of the performance of the CFMM relative to conventional methods for J matrix assembly via split valence basis calculations on 1-d, 2-d and 3-d carbon systems. Our principal conclusions are the following:

(1) The crossover points at which the linear scaling CFMM approach becomes more efficient than conventional quadratic methods are for quite small molecule sizes: approximately 25, 40 and 65 atoms for the 1-d, 2-d, and 3-d model carbon systems with the 3-21G basis. We can estimate the coefficients of linear scaling for the CFMM as being roughly 10^6, 10^7 and 10^8 floating point operations per basis function, for 1-d, 2-d and 3-d systems respectively (although the value for 3-d systems is very rough due to insufficient data). We emphasize that errors in the CFMM are bounded.

(2) The small crossover points, and relatively modest coefficients of linear scaling open the way for DFT calculations on systems containing hundreds and very likely even thousands of atoms in the future. The CFMM is an effective linear scaling solution for the Coulomb problem in DFT. Presuming that existing methods for linear scaling solutions to the problem of iteratively solving for the density in the solid state physics context [21,22] are successfully transferred or extended to molecular DFT calculations, then the dream of linear scaling DFT calculations will have become a reality, as the exchange-correlation contributions to the SCF Hamiltonian are already well-established as scaling linearly [23].

Acknowledgement

This work was supported by Q-Chem Inc, through a Department of Energy SBIR agreement (DE-FG05-93ER81643).

References

[1] L. Greengard and V. Rokhlin, J. Comput. Phys. 60 (1985) 187;
L. Greengard, The rapid evaluation of potential fields in particle systems (MIT Press, Cambridge, 1987).
[2] L. Greengard, Science 265 (1994) 909.
[3] C.A. White and M. Head-Gordon, J. Chem. Phys. 101 (1994) 6593.
[4] K.E. Schmidt and M.A. Lee, J. Stat. Phys. 63 (1991) 1223.
[5] H.Q. Ding, N. Karasawa and W.A. Goddard, J. Chem. Phys. 97 (1992) 4309;
H.Q. Ding, N. Karasawa and W.A. Goddard, Chem. Phys. Letters 196 (1992) 6.
[6] J.P. Dombroski, S.W. Taylor and P.M.W. Gill, J. Phys. Chem., in press.
[7] C.A. White, B.G. Johnson, P.M.W. Gill and M. Head-Gordon, Chem. Phys. Letters 230 (1994) 8.
[8] R. Kutteh, E. Apra and J. Nichols, Chem. Phys. Letters 238 (1995) 173.
[9] M. Challcombe, E. Schwegler and J. Almlöf, Modern developments in Hartree–Fock theory: fast methods for computing

the coulomb matrix (University of Minnesota Supercomputer Institute Research Report UMSI 95/186, 1995). To appear in: Computational chemistry: review of current trends, ed. J. Leczszynski (World Scientific, Singapore, 1996).
[10] M.C. Strain, G.E. Scuseria and M.J. Frisch, Science 271 (1996) 51;
J.C. Burant, M.C. Strain, G.E. Scuseria and M.J. Frisch, Chem. Phys. Letters 248 (1996) 43.
[11] J. Almlöf, K. Faegri Jr. and K. Korsell, J. Comput. Chem. 3 (1982) 385.
[12] H. Horn, H. Weiß, M. Häser, M. Ehrig and R. Ahlrichs, J. Comput. Chem. 12 (1991) 1058.
[13] M. Häser and R. Ahlrichs, J. Comput. Chem. 10 (1989) 104.
[14] P.M.W. Gill, B.G. Johnson and J.A. Pople, Chem. Phys. Letters 217 (1994) 65.
[15] C.A. White and M. Head-Gordon, J. Chem. Phys. 104 (1996) 2620.
[16] G.R. Ahmadi and J. Almlöf, Chem. Phys. Letters 246 (1995) 364.
[17] R.A. Adamson, T.R. Adams and P.M.W. Gill, preprint.
[18] K. Eichkorn, O. Treutler, H. Öhm, M. Häser and R. Ahlrichs, Chem. Phys. Letters 240 (1995) 283, and references therein.
[19] P.M.W. Gill, Advan. Quantum Chem. 25 (1994) 141.
[20] B.G. Johnson, P.M.W. Gill, M. Head-Gordon, C.A. White, D.R. Maurice, R.D. Adamson, T.R. Adams and M. Oumi, Q-Chem (Q-Chem Inc., Pittsburgh, PA, 1995), unpublished.
[21] F. Mauri, G. Galli and R. Car, Phys. Rev. B 47 (1993) 9973
[22] X.-P. Lim, R.W. Nunes and D. Vanderbilt, Phys. Rev. B 47 (1993) 10891.
[23] B.G. Johnson and M.J. Fisch, J. Chem. Phys. 100 (1994) 7429.