FORSCHUNGSZENTRUM JÜLICH GmbH
Zentralinstitut für Angewandte Mathematik
D-52425 Jülich, Tel. (02461) 61-6402

Interner Bericht

An Improved Implementation of the Fast Multipole Method

Holger Dachsel

FZJ-ZAM-IB-2002-14

Dezember 2002

(letzte Änderung: 13.12.2002)

Preprint: Proceedings, 4th MATHMOD Vienna, February 5 - 7, 2003, Wien, 5.2. - 7.2.2003

An Improved Implementation of the Fast Multipole Method

Holger Dachsel
John von Neumann Institute for Computing
Central Institute for Applied Mathematics
Research Centre Jülich
52425 Jülich, Germany
h.dachsel@fz-juelich.de
Phone: +49 2461 616569, Fax: +49 2461 616656

Abstract An improved implementation of the Fast Multipole Method to treat very large systems is described. The implementation is the first step towards the Continuous Fast Multipole Method to evaluate charge distributions in Density Functional and Hartree Fock calculations more efficient.

1. Introduction

In several scientific applications such as molecular dynamics [1] and plasma physics [2] the evaluation of a pairwise potential is required. Very often this is the most time consuming step in a calculation. The direct method to evaluate these potentials scales quadratically with the number of particles N which places a severe restraint on the size of systems which can be treated. Many methods have been proposed to avoid the quadratic scaling [3]. Unfortunately, all these methods lead to unpredictible errors because they rely upon not generally applicable approximations [4]. In particular cut-off approaches show errors which often can not be accepted due to the significance of the long range charge-charge interaction. It is highly desired to avoid the order N^2 scaling. One of the methods to achieve linear scaling is Greengard's [5] Fast Multipole Method (FMM). The purpose of the FMM is to group together remote charges such that a collection of distant charges can be treated as one single charge. The Fast Multipole Method expands local charges in multipole expansions. The multipole expansions of several particles about a common origin can be summed to represent a collection of point charges by just one multipole expansion. The collections of point charges are grouped in boxes which form the FMM tree. The FMM is a computational scheme how to manipulate these expansions to achieve linear scaling. The Fast Multipole Method can be applied to the evaluation of $r^{-n}(n > 0)$ pairwise interactions. Unfortunately, the FMM is not free of parameters. The computation time and the accuracy depend on three parameters, the length of the multipole expansions, the depth of the FMM tree and finally the separation criteria - the number of boxes between two boxes which can interact via multipoles. It is very inconvenient to set the parameters by a user-request. In addition, the three parameters are not independent among each other. One can define a funtion $f(L, D, ws) = 0$, where L ist the length of the expansion, D is the depth of the tree, and ws is the separation criteria. The computation time t depends not only on L, D, and ws. The requested threshold and the kind of distribution, homogeneous or heterogenous has also an impact on the computation time. In our FMM implementation we minimize the computation time $t = t(L, D, ws, kind\,of\,distribution, threshold)$. L, D, and ws are the variables, the

kind of distribution and the *threshold* are the parameters. With this approach we have found a reasonable solution of the problem to separate the particles in near and far field. Within the framework of the HPC-Chem project [6] our implementation of the FMM to treat point charges in a very efficient way is the first step towards the CFMM (Continuous Fast Multipole Method) to calculate charge distributions arising in Density Functional and Hartree Fock calculations. The ideas of FMM can be applied to the evaluation of Electron Repulsion Integrals (ERI's). The computation of the ERI's is in general a step which requires $O(n^4)$ work regarding the number of basis functions n. By several computational techniques [7] the scaling could be improved significantly to $O(n^2)$. The use of CFMM gives the possibility to make a further improvement in scaling, from $O(n^2)$ to $O(n)$. The Coulomb interaction of two charge distributions decreases exponentially with increasing separation, and the two distributions then interact as classical point charges.

2. Theory

The basics of our FMM implementation are described by C. A. White and M. Head-Gordon [8, 9]. In addition, a new scheme of estimating the FMM errors and an approach to evaluate the Wigner rotation matrices [9, 10] more stable for higher multipole moments has been implemented.

A. Factorization of inverse distance

The inverse distance between two point charges located at $\mathbf{a} = (a, \alpha, \beta)$ and $\mathbf{r} = (r, \theta, \phi)$ can be written as an expansion of the associated Legendre polynomials.

$$\frac{1}{|\mathbf{r}-\mathbf{a}|} = \sum_{l=0}^{\infty} P_l(cos(\gamma)) \frac{a^l}{r^{l+1}} \quad (1)$$

$$\frac{1}{|\mathbf{r}-\mathbf{a}|} = \sum_{l=0}^{\infty} \sum_{m=-l}^{m=l} \frac{(l-|m|)}{(l+|m|)} \frac{a^l}{r^{l+1}} P_{lm}(cos(\alpha)) P_{lm}(cos(\theta)) cos(m(\beta-\phi)) \quad (2)$$

$$\frac{1}{|\mathbf{r}-\mathbf{a}|} = \sum_{l=0}^{\infty} \sum_{m=-l}^{m=l} \frac{(l-|m|)}{(l+|m|)} \frac{a^l}{r^{l+1}} P_{lm}(cos(\alpha)) P_{lm}(cos(\theta)) e^{-im(\beta-\phi)} \quad (3)$$

The expansion converges under the condition $a < r$. γ is the angle between the two vectors $\mathbf{a}$ and $\mathbf{r}$. Eq. (2) and Eq. (3) represent a complete factorization of a charge-charge interaction. On the basis on Eq. (3) one can define moments of a multipole expansion. q is the particle charge.

$$\omega_{lm} = qO_{lm} = qa^l \frac{1}{(l+|m|)} P_{lm}(cos(\alpha)) e^{-im\beta} \quad (4)$$

From Eq. (3) one can also define a Taylor expansion.

$$\mu_{lm} = qM_{lm} = q\frac{1}{r^{l+1}}(l-|m|)P_{lm}(cos(\theta))e^{im\phi} \quad (5)$$

Combining Eq. (4) and Eq. (5) together a factorization of the inverse distance can be written in a compact form.

$$\frac{1}{|\mathbf{r}-\mathbf{a}|} = \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\omega_{lm}\mu_{lm} \quad (6)$$

The moments of a multipole expansion and the coefficients of a Taylor expansion about a common origin can of course be summed.

B. Translation operators

Essential to the FMM are the three operators to translate multipole expansions and Taylor expansions in space. The first operator, A, is used to shift a multipole expansion from $\mathbf{a}$ to $\mathbf{a}+\mathbf{b}$.

$$\omega_{lm}(\mathbf{a}+\mathbf{b}) = \sum_{j=0}^{l}\sum_{k=-j}^{j}A_{jk}^{lm}(\mathbf{b})\omega_{jk}(\mathbf{a}) \quad (7)$$

The operator A_{jk}^{lm} is given by

$$A_{jk}^{lm} = O_{l-j,m-k} \quad (8)$$

The second operator, B, transforms a multipole expansion into a Taylor expansion.

$$\mu_{lm} = \sum_{j=0}^{\infty}\sum_{k=-j}^{j}B_{jk}^{lm}\omega_{jk} \quad (9)$$

The operator B_{jk}^{lm} is given by

$$B_{jk}^{lm} = M_{j+l,k+m} \quad (10)$$

The third operator, C, translates a Taylor expansion of a point $\mathbf{r}$ about the origin to a Taylor expansion of $\mathbf{r}$ about a point $\mathbf{a}$.

$$\mu_{lm}(\mathbf{r}-\mathbf{a}) = \sum_{j=0}^{\infty}\sum_{k=-j}^{j}C_{jk}^{lm}(\mathbf{a})\mu_{jk}(\mathbf{r}) \quad (11)$$

The operator C_{jk}^{lm} is given by

$$C_{jk}^{lm} = O_{j-l,k-m} \quad (12)$$

3. The Fast Multipole Method

The FMM consists of several parts. First all particles are enclosed by a box with coordinate ranges [0,1]x[0,1]x[0,1]. The parent box which contains all the particles is divided in half along each Cartesian axis to yield a set of 8 smaller child boxes. The child boxes are subdivided again. The depth of the tree is determined so that the computation time becomes a minimum in achieving an error in the energy which is less or equal to a requested threshold. The particles are sorted by box numbers using the radix sort algorithm [11] which scales linearly. In addition to scaling and sorting the FMM consists of four passes schematically shown in Fig. (1). In pass 1 the charges contained within each lowest level box are expanded in multipoles about the center of the box. The multipole expansions are translated to the center of the parent boxes. In pass 2 the multipole expansions are transformed into Taylor expansions. The two boxes must be separated by at least one box on the current level on the tree, but only provided that parents of the two boxes are not separated on the next higher level on the tree. Pass 2 is by far the most time comsuming step of the FMM. In pass 3 the parent's Taylor expansions are translated to the centers of the parent's children. At the end of pass 3 each lowest level box contains a Taylor expansion of all far field interactions. In pass 4 for each lowest level box the multipole expansion and the Taylor expansion are multiplied. The sum over all lowest level boxes gives the far field energy. Finally, in pass 5 the remaining near-field energy is computed by the direct method.

4. The Wigner Rotation Matrices

The conventional Fast Multipole Method requires $O(L^4)$ work with regard to the length of the multipole expansion L. $O(L^3)$ scaling can be achieved by performing the translations in three steps. First the moments of a multipole expansion or the coefficients of a Taylor expansion are rotated about the z-axis and y-axis such that the phase factors in Eq. (4) and Eq. (5) vanish and the associated Legendre polynomials P_{lm} degenerate to the Legendre polynomials P_l. In the second step the translations are performed.

$$\omega_{lm} = \sum_{j=m}^{l} \frac{a^{l-j}}{(l-j)!} \omega_{jm} \quad (13)$$

$$\mu_{lm} = \sum_{j=m}^{\infty} \frac{(j+l)!}{r^{j+l+1}} \omega_{j,-m} \quad (14)$$

$$\mu_{lm} = \sum_{j=l}^{\infty} \frac{a^{j-l}}{(j-l)!} \mu_{jm} \quad (15)$$

Finally, the translated multipole moments and the Taylor coefficients are rotated back using the inverse rotation matrices. The rotation about the z-axis is simply a complex multiplication. The only difficult portion is the determination of the Wigner rotation

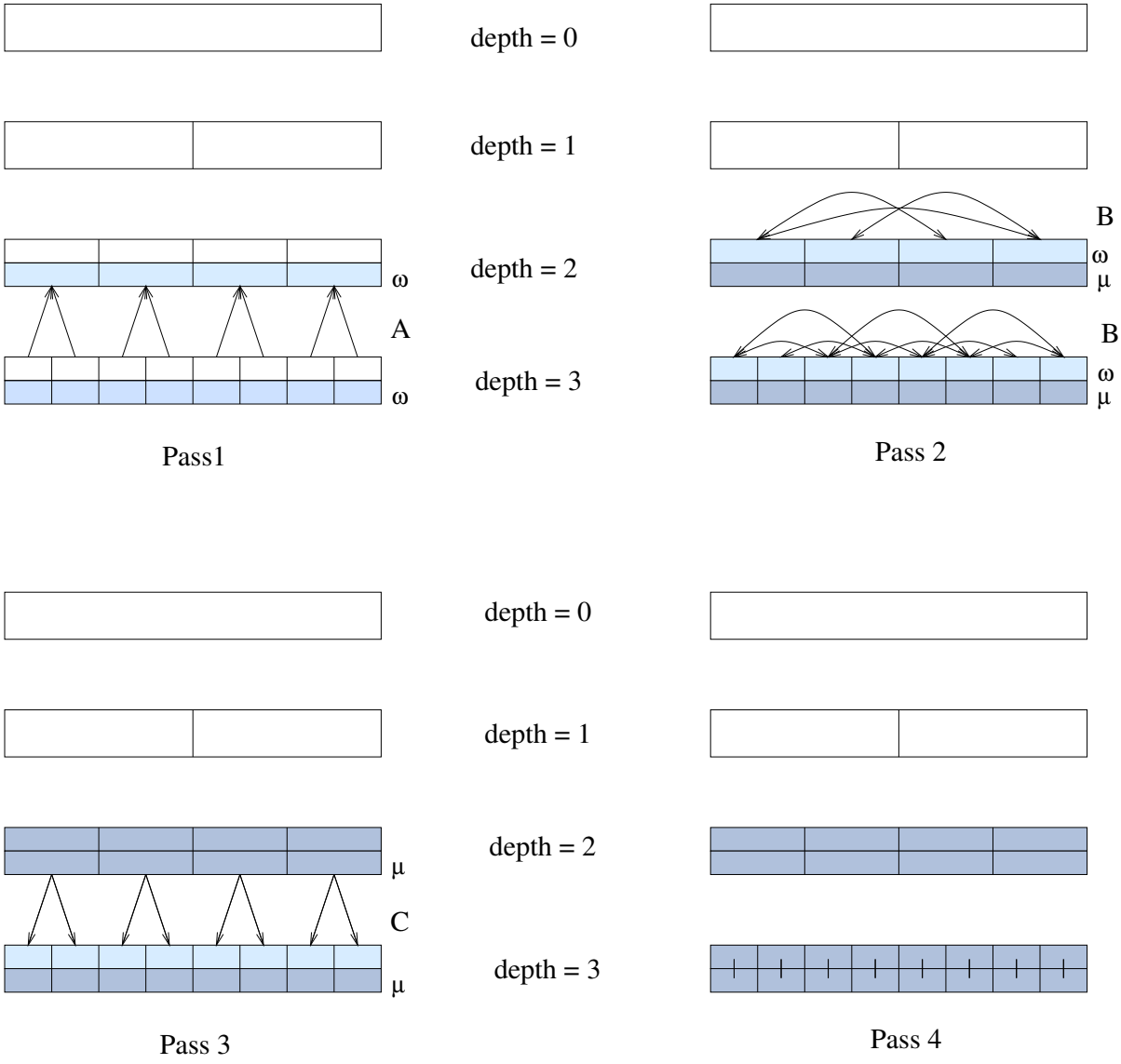

Fig. (1): Schematic view on one dimensional FMM with parameter $ws = 1$

matrices $d^l_{km}(\theta)$ which correspond to the rotation about the y-axis. The analytical calculation of the $d^l_{km}(\theta)$ requires $O(L^4)$ work and is numerically instable.

$$d^l_{k\,m} = \frac{1}{2^l}\sqrt{\frac{(l-m)!\,(l+m)!}{(l-k)!\,(l+k)!}}\,(1+sign(k)cos(\theta))^{|k|}\,(sin(\theta))^{m-|k|}$$

$$\sum_{n=0}^{l-m}(-1)^{l-m-n}\binom{l-k}{n}\binom{l+k}{l-m-n}\,(1+cos(\theta))^n\,(1-cos(\theta))^{l-m-n} \quad (16)$$

$$d^l_{m\,k} = (-1)^{k+m}d^l_{k\,m} \quad (17)$$

$$l \geq 0 \quad , \quad k = -l, ..., l \quad , \quad |k| \leq m \leq l \quad (17a)$$

$$d^l_{k\,m} = (-1)^{k+m} d^l_{-k\,-m} \quad (18)$$

$$l > 0 \quad , \quad m = -l, ..., l-1 \quad , \quad k = -l, ..., -(m+1) \quad (18a)$$

The essential recursion relation we will use to determine the rotation matrices is given by White [9] and Edmonds [10].

$$d^l_{k+1\,m} = \frac{k+m}{\sqrt{l(l+1)-k(k+1)}} \frac{sin(\theta)}{1+cos(\theta)} d^l_{k\,m}$$

$$+\sqrt{\frac{l(l+1)-m(m-1)}{l(l+1)-k(k+1)}} d^l_{k\,m-1} \quad (19)$$

$$d^l_{0m} = \sqrt{\frac{(l-m)!}{(l+m)!}} P_{lm}, \ m \geq 0 \quad (19a)$$

$$d^l_{0m} = (-1)^m \sqrt{\frac{(l-|m|)!}{(l+|m|)!}} P_{l|m|}, \ m < 0 \quad (19b)$$

Unfortunately, Eq. (19) becomes instable in case of higher moments. We have combined Eq. (19) with a second recurrence to overcome the numerical instabilities.

$$d^l_{k\,m-1} = \sqrt{\frac{l(l+1)-k(k+1)}{l(l+1)-m(m-1)}} d^l_{k+1\,m}$$

$$-\frac{k+m}{\sqrt{l(l+1)-m(m-1)}} \frac{sin(\theta)}{1+cos(\theta)} d^l_{k\,m} \quad (20)$$

$$d^l_{kl} = \frac{1}{2^l} \sqrt{\frac{(2l)!}{(l-k)!(l+k)!}} (sin(\theta))^{l-k} (1+cos(\theta))^k \quad (20a)$$

In addition to the two recurrences the error accumulations are evaluated for both of the recurrences to decide which recursion relation is more accurate for a given component of the rotation matrix. Both of the recursion relations should be used only for $cos(\theta) \geq 0$. In case $cos(\theta) < 0$ addition theorems can be used given by Edmonds [10]. The combination of the two recurrences show a significant improvement of accuracy. Tab. (1) shows the absolute errors for $\theta = \frac{\pi}{2}$.

L	First recursion relation	Both recursion relations
5	$1.20 \cdot 10^{-15}$	$1.11 \cdot 10^{-16}$
10	$3.33 \cdot 10^{-14}$	$2.78 \cdot 10^{-16}$
15	$1.65 \cdot 10^{-12}$	$7.49 \cdot 10^{-16}$
20	$1.99 \cdot 10^{-10}$	$1.50 \cdot 10^{-15}$
25	$1.35 \cdot 10^{-8}$	$6.27 \cdot 10^{-15}$
30	$1.07 \cdot 10^{-6}$	$6.89 \cdot 10^{-14}$
35	$9.64 \cdot 10^{-5}$	$1.38 \cdot 10^{-13}$
40	$8.89 \cdot 10^{-3}$	$1.35 \cdot 10^{-12}$
45	$5.52 \cdot 10^{-1}$	$4.48 \cdot 10^{-12}$
50	$1.18 \cdot 10^{2}$	$3.13 \cdot 10^{-11}$
55	$9.63 \cdot 10^{3}$	$1.27 \cdot 10^{-10}$
60	$1.54 \cdot 10^{6}$	$6.04 \cdot 10^{-10}$
65	$9.79 \cdot 10^{7}$	$4.83 \cdot 10^{-9}$

Tab. (1): Maximum absolute errors in computation of the d_{km}^{l}

5. Error estmation

The error estimation by White and Head-Gordon [8] gives an upper limit for the error but is often not practible. We have used a different approach. The FMM has two error sources, the truncation of the multipole expansions and the truncation in the transformation of the multipole moments to Taylor expansions.

$$2\,\Delta E = \sum_{ilevel} \sum_{ibox(ilevel)} \sum_{jbox(ibox)} \sum_{I(ibox)=1}^{N(ibox)} \sum_{J(jbox)=1}^{N(jbox)} q_I q_J \Delta\left(\frac{1}{r_{IJ}}\right) \quad (21)$$

$$2\,\Delta E = C\,\Delta\left(\frac{1}{r}\right) \quad (22)$$

$\mathbf{q}$ is the vector of charges. C depends also on the kind of distribution.

$$\Delta\left(\frac{1}{r}\right) = \Delta_{truncation}\left(\frac{1}{r}\right) + \Delta_{transformation}\left(\frac{1}{r}\right) \quad (23)$$

$\Delta_{truncation}\left(\frac{1}{r}\right)$ and $\Delta_{transformation}\left(\frac{1}{r}\right)$ are given by

$$\Delta_{truncation} = \sum_{l=0}^{L} \sum_{m=-l}^{l} \sum_{j=L+1}^{\infty} \sum_{k=-j}^{j} I_{lmjk} \quad (24)$$

$$\Delta_{transformation} = \sum_{l=L+1}^{\infty} \sum_{m=-l}^{l} \sum_{j=0}^{\infty} \sum_{k=-j}^{j} I_{lmjk} \quad (25)$$

The components I_{lmjk} are defined by

$$I_{lmjk} = \sqrt{\frac{(j-k+l-m)!(j+k+l+m)!}{(j-k)!(j+k)!(l-m)!(l+m)!}}$$

$$\int_0^1 \int_0^1 \int_0^1 r^j |d_{0m}^j| \, dx \, dy \, dz \cdot \int_0^1 \int_0^1 \int_0^1 r^l |d_{0m}^l| \, dx \, dy \, dz \quad (26)$$

Tab. (2) shows the number of multipoles depending on a requested threshold.

Req. abs. error	Abs. error	L
10^{-2}	$0.47 \cdot 10^{-2}$	1
10^{-3}	$0.98 \cdot 10^{-3}$	3
10^{-4}	$0.99 \cdot 10^{-4}$	5
10^{-5}	$0.26 \cdot 10^{-5}$	7
10^{-6}	$0.39 \cdot 10^{-6}$	10
10^{-7}	$0.76 \cdot 10^{-7}$	13
10^{-8}	$0.21 \cdot 10^{-8}$	17
10^{-9}	$0.29 \cdot 10^{-9}$	21
10^{-10}	$0.66 \cdot 10^{-10}$	25
10^{-11}	$0.93 \cdot 10^{-11}$	30
10^{-12}	$0.38 \cdot 10^{-12}$	34
10^{-13}	$0.66 \cdot 10^{-13}$	39

Tab. (2): Number of multipole moments depending on requested absolute errors

6. Implementation issues

Our FMM implementation is designed to evaluate systems consisting of billions of point charges. Heterogeneous distributions can be treated in the same efficient way as homogeneous distributions. The parameters of the FMM are determined such that the computation time is minimal depending on a requested threshold of the energy. All empty boxes on the tree are completely neglected. Any arrays in the dimension of all boxes are avoided. The maximal depth of the trees depends only on the integer length. Logical bit operations are used for box numbering. A logical right shift by three positions of a box number results in the number of the parent box. A logical left shift by three positions of a box number gives the number range of all child boxes. Our FMM implementation is fully based on spherical coordinates.
We have made the first approach in parallelizing our FMM implementation. An efficiency of more than 90% up to 16 CPU's have been seen. In the parallel version Pass 3 is avoided because it can not be parallelized efficiently. It is easily possible to shift the work which is done in Pass 3 to Pass 4. All the remaining four passes are parallelized. Our parallelization strategy is based on the replicated data model which is a severe bottleneck. We will implement a data distributed version.

7. Summary and Outlook

We have described an improved implementation of the rotation based Fast Multipole Method to evaluate systems of point charges as the basis for the Continuous Fast Multipole Method to treat charge distributions. First steps in parallelizing the program have been made. Further work to improve the parallel performance is necessary.
The serial version of our FMM program is able to traet very large systems of point charges up to several billions of particles. We have proposed a new approach for the separation of near and far field within the theory of FMM to minimize the computation time depending on an user-requested threshold.

References

[1] M. P. Allen and D. J. Tildesley, Computer Simulation of Liquids (Oxford University, Oxford, 1990)

[2] J. M. Dawson, Rev. Mod. Phys. **55**, 403 (1983)

[3] W. E. van Gunsternen and H. J. C. Berendsen, Angew. Chem. Int. Ed. Engl. **29**, 992 (1990)

[4] M. Saito, Mol. Simul. **8**, 321 (1992)

[5] L. Greengard and V. Rokhlin, J. Comput. Phys. **60**, 187 (1985)

[6] http://www.fz-juelich.de/hpc-chem

[7] J. M. Ugalde and C. Sarasola, Int. J. of Quantum Chemistry **62**, 273 (1997)

[8] C. A. White and M. Head-Gordon, J. Chem. Phys. **101**, 6593 (1994)

[9] C. A. White and M. Head-Gordon, J. Chem. Phys. **105**, 5061 (1996)

[10] A. R. Edmonds, Angular Momentum in Quantum Mechanics (Princeton University Press, Princeton, 1957)

[11] I. J. Davis, Computer J. **35**, 636 (1992)