The Ewald sum on the FPS/164 MAX

M. Leslie.

1 Introduction

Much has been written before about the Ewald sum and its application to both static and dynamic simulations [1] [2]. This article will therefore not attempt to repeat the theory. The aim is to show how the reciprocal space part of the sum may be reformulated such that the bulk of the floating point arithmetic is carried out as matrix multiplication operations. This allows the very fast matrix multiplication software provided with the FPS/164 MAX to be used. Although written with the FPS in mind, the method described is useful for any architecture where matrix multiplication may be performed at a much faster rate than ordinary FORTRAN. Overall the reciprocal space summation runs at 40 MFLOPS on the FPS/164 with 3 MAX boards at Daresbury. This should be compared with 200 MFLOPS on the CRAY XMP. The direct space part of the sum is slower on the 164 by a factor of 20 compared with the XMP. The MAX boards cannot be used for this part of the sum. However, it is possible to adjust the Ewald parameter α such that the bulk of the arithmetic is carried out in the reciprocal space sum with a larger reciprocal space cutoff and a smaller direct space cutoff. The direct space part of the sum will not be discussed further here.

The expression for the reciprocal space part of the energy and forces on a particle i is , following [1] equation (11b):

$$\phi_i = \sum_{\mathbf{k}} A(\mathbf{k}) \Re[(\sum_j q_j e^{i\mathbf{k}\bullet\mathbf{r}_j}) q_i e^{-i\mathbf{k}\bullet\mathbf{r}_i}] \tag{1}$$

$$-\mathbf{F}_i = \sum_{\mathbf{k}} A(\mathbf{k}) \mathbf{k} \Im[(\sum_j q_j e^{i\mathbf{k}\bullet\mathbf{r}_j}) q_i e^{-i\mathbf{k}\bullet\mathbf{r}_i}] \tag{2}$$

Where:

$$A(\mathbf{k}) = \frac{\pi}{\alpha^2 V} e\left(-\frac{k^2}{4\alpha^2}\right) / \frac{k^2}{4\alpha^2} \tag{3}$$

$$\phi = \frac{1}{2} \sum_i \phi_i$$

The notation follows [1] except that V is the unit cell volume. The sum over $\mathbf{k}$ need only be carried out over half of reciprocal space and the final result doubled since reciprocal space is always centrosymmetric. The z component of the reciprocal lattice vector will always be taken as positive.

In general, the unit cell may have any shape and in the above expressions $\mathbf{k}$ is $2\pi\times$ a reciprocal lattice vector in orthonormal coordinates and

$\mathbf{r}$ is an atom position in orthonormal coordinates. It is more convenient to work in a coordinate system defined relative to the unit cell lattice vectors. Let $\mathbf{L}$ be a matrix in the orthonormal coordinate system whose columns are the unit cell lattice vectors. Then the coordinates of an atom $\mathbf{R}$ in the non-orthogonal crystallographic reference frame may be defined as

$$\mathbf{r} = \mathbf{L} \bullet \mathbf{R}$$

The reciprocal lattice vectors in the new reference frame transform to

$$\mathbf{k} = \mathbf{K} \bullet \mathbf{L}^{-1}$$

where $\mathbf{K}$ is now 2π times an integer vector. Hence

$$\mathbf{k} \bullet \mathbf{r} = \mathbf{K} \bullet \mathbf{L}^{-1} \bullet \mathbf{L} \bullet \mathbf{R} = \mathbf{K} \bullet \mathbf{R}$$

Equations (1) and (2) may be written in terms integer reciprocal lattice vectors and non-orthogonal atom positions.

$$\phi_i = \sum_{\mathbf{k}} A(\mathbf{k}) \Re[(\sum_j q_j e^{i\mathbf{K} \bullet \mathbf{R}_j}) q_i e^{-i\mathbf{K} \bullet \mathbf{R}_i}] \tag{4}$$

$$-\mathbf{F}_i = \sum_{\mathbf{k}} A(\mathbf{k}) \mathbf{k} \Im[(\sum_j q_j e^{i\mathbf{K} \bullet \mathbf{R}_j}) q_i e^{-i\mathbf{K} \bullet \mathbf{R}_i}] \tag{5}$$

2 Description of method

The calculation is divided into seven sections.

2.1 Step 1. Calculate $A(k)$

The $A(\mathbf{k})$ factors are calculated according to (3). Note that these do not depend on the positions of the atoms in the unit cell, so only need be calculated once if the unit cell shape is not allowed to change.

2.2 Step 2. Factorization

The second step is to factorize the complex exponential. This greatly reduces the computation time since the factors may be evaluated by complex multiplication rather than a complex exponentiation.

$$q_j e^{i\mathbf{K} \bullet \mathbf{R}} = q_j e^{iK_x R_x} e^{iK_y R_y} e^{iK_z R_z}$$

The complex exponentials $e^{2\pi i R_x}$, $e^{2\pi i R_y}$ and $e^{2\pi i R_z}$ must be calculated. Then the higher values of K are calculated by complex multiplication.

$$e^{2(2\pi i R_x)} = e^{2\pi i R_x} \times e^{2\pi i R_x}$$

If the maximum reciprocal lattice vectors in the x, y and z directions are N_x, N_y and N_z respectively and there are M atoms in the unit cell, the factors are stored in three matrices $\mathbf{V}_x$, $\mathbf{V}_y$ and $\mathbf{V}_z$. $\mathbf{V}_x$ is a M by $2N_x$ matrix with columns alternately the real and imaginary parts of the complex products. The $K_x = 0$ vector is not stored. The elements of $\mathbf{V}_x$ are multiplied by q_j. $\mathbf{V}_y$ is a M by $2N_y$ matrix. $\mathbf{V}_z$ is a M by $2N_z + 1$ matrix. Column 1 of $\mathbf{V}_z$ is the real part of the $K_z = 0$ vector and has all elements set to 1.

2.3 Step 3. Calculation of the xy product matrix

The complex product of the factors $e^{iK_xR_x}e^{iK_yR_y}$ is now calculated and stored for all K_x and K_y. In this calculation allowance is made for $K_x = 0$ and $K_y = 0$, as well as calculating the complex conjugate product.

1 **Row** The real part of $K_x = 0$, $K_y = 0$ is set equal to q_i.

$2N_y$ **Rows** These rows are the transpose of $\mathbf{V}_y$ multiplied by q_i, and are for $K_x = 0$.

$2N_x$ **Rows** These rows are the transpose of $\mathbf{V}_x$.

$4N_xN_y$ **Rows** These rows are the complex product of two non-zero K_x and K_y vectors followed by the complex conjugate product. The products are calculated in the same DO loop since they have the same multiplications in them. This gives a matrix $\mathbf{V}_4$ of dimension $(1+2N_x)(1+2N_y)$ by M.

2.4 Step 4. First matrix product.

The matrix product $\mathbf{V}_4\mathbf{V}_z$ gives a matrix $\mathbf{W}$ whose elements contain the real and imaginary parts of $\sum_j q_j e^{i\mathbf{K}\bullet\mathbf{R}}$. $\mathbf{W}$ is a $(1+2N_x)(1+2N_y)$ by $(1+2N_z)$ matrix. Henceforth the following notation will be used:

C_{xy}	is the real part of	$\mathbf{V}_4$
S_{xy}	is the imaginary part of	$\mathbf{V}_4$
C_z	is the real part of	$\mathbf{V}_z$
S_z	is the imaginary part of	$\mathbf{V}_z$

$$\begin{aligned}
&\sum_j (C_{xy}+iS_{xy})\times(C_z+iS_z)\\
&\quad= \sum_j C_{xy}C_z - \sum_j S_{xy}S_z + i\times(\sum_j C_{xy}S_z + \sum_j S_{xy}C_z)\\
&\quad= W_{CC} - W_{SS} + i\times(W_{CS}+W_{SC})
\end{aligned}$$

The product of the complex conjugate of the xy vector with z will also be needed, giving the reciprocal lattice vector $-x-yz$.

$$\begin{aligned}
&\sum_j (C_{xy}-iS_{xy})\times(C_z+iS_z)\\
&\quad= \sum_j C_{xy}C_z + \sum_j S_{xy}S_z + i\times(\sum_j C_{xy}S_z - \sum_j S_{xy}C_z)\\
&\quad= W_{CC} + W_{SS} + i\times(W_{CS}-W_{SC})
\end{aligned}$$

Equations (4) and (5) may be rewritten using this notation:

$$\begin{aligned}
\phi_i &= \sum_{xyz} A(xyz)\Re\{[(W_{CC}-W_{SS})+i\times(W_{CS}+W_{SC})]\times\\
&\qquad [(C_{xy}-iS_{xy})\times(C_z-iS_z)]\}
\end{aligned}\tag{6}$$

$$\begin{aligned}
-\mathbf{F}_i &= \sum_{xyz} A(xyz)\mathbf{k}(xyz)\Im\{[(W_{CC}-W_{SS})+i\times(W_{CS}+W_{SC})]\times\\
&\qquad [(C_{xy}-iS_{xy})\times(C_z-iS_z)]\}
\end{aligned}\tag{7}$$

2.5 Step 5. Setup for second matrix product

Equations (6) and (7) are now rearranged so that another matrix product may be carried out. First the terms in xyz and $-x-yz$ are written explicitly.

$$
\begin{aligned}
\phi_i &= \sum_{xyz,z>0} \Re\{\{A(xyz)[(W_{CC}-W_{SS})+i(W_{CS}+W_{SC})]\times \\
&\quad (C_{xy}-iS_{xy})+A(-x-yz)[(W_{CC}+W_{SS})+i(W_{CS}-W_{SC})]\times \\
&\quad (C_{xy}+iS_{xy})\}\times(C_z-iS_z)\} \\
&= \sum_{z,z>0}\sum_{xy}\Re\{\{ \\
&\quad [A(xyz)(W_{CC}-W_{SS})+A(-x-yz)(W_{CC}+W_{SS})]\times C_{xy} \\
&\quad +[A(xyz)(W_{CS}+W_{SC})-A(-x-yz)(W_{CS}-W_{SC})]\times S_{xy} \\
&\quad +[A(xyz)(W_{CS}+W_{SC})+A(-x-yz)(W_{CS}-W_{SC})]\times iC_{xy} \\
&\quad +[-A(xyz)(W_{CC}-W_{SS})+A(-x-yz)(W_{CC}+W_{SS})]\times iS_{xy}\} \\
&\quad \times(C_z-iS_z)\} \\
&= \sum_{z,z>0}\sum_{xy}\Re\{\{X_{CC}C_{xy}+X_{CS}S_{xy}+iX_{SC}C_{xy}+iX_{SS}S_{xy}\} \\
&\quad \times(C_z-iS_z)\} \qquad (8)
\end{aligned}
$$

$$
\begin{aligned}
-\mathbf{F}_i &= \sum_{z,z>0}\sum_{xy}\Im\{\{\mathbf{Y}_{CC}C_{xy}+\mathbf{Y}_{CS}S_{xy}+i\mathbf{Y}_{SC}C_{xy}+i\mathbf{Y}_{SS}S_{xy}\} \\
&\quad \times(C_z-iS_z)\} \qquad (9)
\end{aligned}
$$

Where:

$$
\begin{aligned}
\mathbf{Y}_{CC} &= A(xyz)\mathbf{k}(xyz)(W_{CC}-W_{SS})+ \\
&\quad A(-x-yz)\mathbf{k}(-x-yz)(W_{CC}+W_{SS}) \\
\mathbf{Y}_{CS} &= A(xyz)\mathbf{k}(xyz)(W_{CS}+W_{SC})- \\
&\quad A(-x-yz)\mathbf{k}(-x-yz)(W_{CS}-W_{SC}) \\
\mathbf{Y}_{SC} &= A(xyz)\mathbf{k}(xyz)(W_{CS}+W_{SC})+ \\
&\quad A(-x-yz)\mathbf{k}(-x-yz)(W_{CS}-W_{SC}) \\
\mathbf{Y}_{SS} &= -A(xyz)\mathbf{k}(xyz)(W_{CC}-W_{SS})+ \\
&\quad A(-x-yz)\mathbf{k}(-x-yz)(W_{CC}+W_{SS})
\end{aligned}
$$

In the above equations, allowance must be made for the terms for which x, y or z are zero.

2.6 Step 6. Second matrix product

The elements of X and Y in equations (8) and (9) are written into one large matrix $\mathbf{V}_5$. For each pair of reciprocal lattice vectors $\mathbf{k}(xyz)$ and $\mathbf{k}(-x-yz)$ 8 elements of the array $\mathbf{V}_5$ are written. These are shown below, together with two elements from the array $\mathbf{V}_4$.

$$\begin{array}{rrl} X_{CC} & X_{CS} & C_{xy} \\ X_{SC} & X_{SS} & S_{xy} \\ Y_{SCx} & Y_{SSx} & \\ -Y_{CCx} & -Y_{CSx} & \\ Y_{SCy} & Y_{SSy} & \\ -Y_{CCy} & -Y_{CSy} & \\ Y_{SCz} & Y_{SSz} & \\ -Y_{CCz} & -Y_{CSz} & \end{array}$$

The following notation will now be used

$$\begin{array}{lcl} V_{511} = X_{CC} & \text{or} & Y_{SC\alpha} \\ V_{512} = X_{CS} & \text{or} & Y_{SS\alpha} \\ V_{521} = X_{SC} & \text{or} & -Y_{CC\alpha} \\ V_{522} = X_{CC} & \text{or} & -Y_{CS\alpha} \end{array}$$

Where $\alpha = x$, y or z. Fewer elements will need to be written for the special cases of $x = 0$, $y = 0$ and $z = 0$. It may be shown that the array $\mathbf{V}_5$ has dimensions $4(1 + 2N_z)$ by $(1 + 2N_x)(1 + 2N_y)$. The matrix product $\mathbf{V}_5\mathbf{V}_4 = \mathbf{V}_6$ then gives the sum over xy in equations (8) and (9). The matrix product is carried out as one large product rather than four smaller products as this reduces the overheads on the FPS associated with loading the MAX boards with one of the matrices.

2.7 Step 7. Final scalar product

The final step is to carry out a scalar product over all of the real and imaginary parts of $\mathbf{V}_6$ with $\mathbf{V}_z$. The following notation will be used.

$$\begin{aligned} V_{6C} &= \sum_{xy}[V_{511}C_{xy} + V_{512}S_{xy}] \\ V_{6S} &= \sum_{xy}[V_{521}C_{xy} + V_{522}S_{xy}] \end{aligned}$$

Equations (8) and (9) now become:

$$\begin{aligned} \phi_i &= \Re\sum_z(V_{6C} + iV_{6S})(C_z - iS_z) \\ &= \sum_z V_{6C}C_z + \sum_z V_{6S}S_z \qquad (10) \\ -F_i &= \Im\sum_z(iV_{6C} - V_{6S})(C_z - iS_z) \\ &= \sum_z V_{6C}C_z + \sum_z V_{6S}S_z \qquad (11) \end{aligned}$$

3 Cutoffs

The choice of Ewald parameter α determines the cutoffs used in the direct and reciprocal space sums. The expressions used are taken from [3]. Let

A be an accuracy parameter with the direct and reciprocal space sums calculated to this accuracy. Then

$$f = (-\ln A)^{1/2}$$

The direct space cutoff is $r_{max} = f/\alpha$ and the reciprocal space cutoff is $k_{max}/2\pi = \alpha f/\pi$. The value of α chosen to give approximately equal numbers of terms in the two series is:

$$\alpha = (M\pi^3/V^2)^{1/6}$$

However, this value of α may be changed to make the reciprocal space cutoff larger and the direct space cutoff smaller. The maximum reciprocal lattice vectors in the x, y and z directions N_x, N_y and N_z are determined from the formula given below.

$$N_x = (k_{max}/2\pi)/|\mathbf{l}_x|$$

where $\mathbf{l}_x$ is the x column of the $\mathbf{L}$ matrix defined in the introduction.

Usually all vectors outside the spherical cutoff are omitted from the calculation. The method as described so far includes all vectors within a cube (parallelepiped in general). Hence the number of vectors calculated is approximately twice the number within the spherical cutoff. This may be improved by omitting those lines in reciprocal space in the z direction which do not intersect the cutoff sphere. The ratio of the total number of points calculated to the number in the cutoff sphere is now reduced to 1.5. The cutoff is applied in step 3 above, the row dimension of the matrix $\mathbf{V}_4$ will be less than $(1 + 2N_x)(1 + 2N_y)$.

4 Timings

The times given below are for 1000 basis atoms with a reciprocal space cutoff $N_x = N_y = N_z = 9$. This gives 3429 reciprocal lattice vectors within the cube. Of these, 1559 are within the spherical cutoff. There are 181 lines of reciprocal space points in the z direction, of which 131 intersect the cutoff sphere. This gives a total of 2489 reciprocal space points calculated by the matrix method.

Timings on FPS and XMP in seconds				
STEP		FPS	XMP	FPS/XMP
1		0.051	0.0016	32.0
2		0.153	0.0031	50.0
3		0.218	0.0056	39.0
4	MAX	0.211	0.0493	4.3
5		0.019	0.0020	10.0
6	MAX	0.650	0.1977	3.3
7		0.053	0.0016	33.0
Total		1.354	0.2650	5.3

The total time on the XMP using the full spherical cutoff without any matrix multiplication was 0.330 s. This is slightly greater than the matrix multiplication time despite there being fewer vectors calculated. Steps 2, 3 and 7 are all memory limited on the FPS. They could all be speeded up by a factor of 2-3 if the table memory were used, but this part would need to be programmed in assembler. On the XMP, vector loops can carry out three vector memory references simultaneously, so the loops are not memory limited.

References

[1] N. Anastasiou and D. Fincham Comput. Phys. Comm. 25 (1982) 159

[2] W. Smith. CCP5 Newsletter 21 (1986) 37

[3] C. R. A. Catlow and W. C. Mackrodt in Computer simulation of solids. Lecture notes in Physics Vol. 166. Berlin Springer-Verlag (1982) C. R. A. Catlow (ed.)