Acta Crystallographica Section A
Foundations of Crystallography
ISSN 0108-7673

Received 20 July 2001
Accepted 25 July 2001

An efficient *ab initio* calculation of powder diffraction intensity using Debye's equation

Richard F. Grover and David R. McKenzie*

Department of Applied Physics, School of Physics A28, The University of Sydney, Sydney, NSW 2006, Australia. Correspondence e-mail: mckenzie@physics.usyd.edu.au

The Debye equation gives the spherically averaged diffracted intensity from a group of atoms and is exact under the first Born, or kinematic, approximation. Algebraic simplifications are developed for calculating multiplicities in the double summation and are used in a new algorithm for implementing this equation. The results for cubic, body-centred cubic and face-centred cubic systems agree exactly with previous methods while achieving substantial computational advantage.

1. Powder diffraction with Debye's equation

Debye's equation for the intensity diffracted by an array of atoms, taking all orientations with equal probability, was published in 1915 (Debye, 1915) and is normally written

$$I(Q) = \sum_i \sum_j f_i f_j \operatorname{sinc}(Qr_{ij}), \qquad (1)$$

where $\operatorname{sinc}(x) = \sin(x)/x$. The scattering factors, f_i and f_j, are in general functions of $Q = 4\pi\sin(\theta)/\lambda$ but, in the case of neutron diffraction, are constants termed *scattering lengths*. This equation is exact in the first Born, or kinematic, approximation and deals explicitly with the small-angle scattering as well as the scattering to the Bragg positions. The equation can be applied to powders that show no preferred orientation. The summation of (1) extends over all atoms i and j in the array, with the value of $\operatorname{sinc}(Qr_{ij})$ equal to unity for self terms ($i = j$). A derivation of the Debye equation can be found in Warren (1990).

In this communication, we present an efficient algorithm for calculating the diffracted intensity *ab initio* from the Debye equation, based on an algebraic solution of equation (1), which provides substantial computational advantage over methods of direct summation. An efficient algorithm for the computation of this equation under general circumstances may allow its use for the fitting of powder patterns, possibly as an alternative to the Rietveld method (Rietveld, 1967).

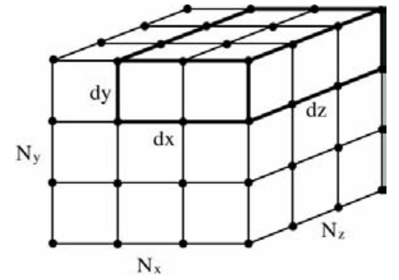

Figure 1
Simple cubic crystallite showing the arbitrary calculation cell of extent d_x, d_y and d_z in the x, y and z directions.

It is known that the number of terms in the summation of the Debye equation can be considerably reduced by noting that it can be written in terms of a *distance multiplicity function*, $M(r)$ (Marciniak *et al.*, 1996). This function effectively evaluates all interatomic distances within the crystallite and counts the number of pairs separated by each distance. Using this approach, we can rewrite the Debye equation as

$$I(Q) = \sum_{r_{ij}} M(r_{ij}) f_i f_j \operatorname{sinc}(Qr_{ij}), \qquad (2)$$

where summation is over all possible interatomic separation distances. With the approach of Marciniak *et al.* (1996), in which the distance from each atom to all others is calculated directly, the evaluation of $M(r_{ij})$ for a system containing n atoms requires $n(n-1)$ operations, growing approximately as the fifth power of the number of unit cells, N (Marciniak *et al.*, 1996).

Taking the structure shown in Fig. 1, we can reduce the calculation of $M(r_{ij})$ to a single evaluation for each possible interatomic separation, which scales, at worst, as the third power of the system size. To determine the value of the multiplicity function for this distance, we note that each cell will contain four body diagonals, giving a count of eight such separations. If the cell has a zero extent in any one direction, however, then there are only four separations at this distance. It is easily seen that this extends to two separations with only one non-zero extent and unity when all are zero, giving 2^α, where α is the number of d_x, d_y and d_z that are non-zero. The remainder of the form of the multiplicity function comes from determining the number of possible arrangements for which a cell of the given extent fits within the crystallite. It can be readily verified from Fig. 1 that the final form is

$$M(d_x, d_y, d_z) = 2^\alpha (N_x + 1 - d_x)(N_y + 1 - d_y)(N_z + 1 - d_z). \qquad (3)$$

Now, the body-centred cubic (b.c.c.) lattice essentially consists of a second simple cubic lattice displaced from the first by half the unit cell body diagonal. Thus, we may use (3) directly to evaluate the multiplicity for separations within either of these two lattices. This leaves only the separations between the lattices to be evaluated, which are only relevant when the cell has a non-zero extent in all directions. Evaluating the cell multiplicity and distance gives

$$M(d_x, d_y, d_z) = 16(N_x + 1 - d_x)(N_y + 1 - d_y)(N_z + 1 - d_z) \qquad (4)$$

$$r(d_x, d_y, d_z) = [(d_x - \tfrac{1}{2})^2 + (d_y - \tfrac{1}{2})^2 + (d_z - \tfrac{1}{2})^2]^{1/2}. \qquad (5)$$

It is possible to arrange the evaluation of equations (3), (4) and (5) in a single loop over the allowable values of d_x, d_y and d_z, giving a solution scaling as the number of unit cells, $N = N_xN_yN_z$. Following a method exactly analogous to that of the b.c.c. case allows the evaluation of the face-centred cubic (f.c.c.) crystal structure. Considering the same arbitrary cell as for the basic cubic lattice, we may determine the separations for atoms within each of these lattices. Furthermore, the four expressions can be combined into a single expression as

$$\begin{aligned} M(d_x, d_y, d_z) = 2^{\alpha}[&(N_x + 1 - d_x)(N_y + 1 - d_y)(N_z + 1 - d_z) \\ &+ (N_x - d_x)(N_y - d_y)(N_z + 1 - d_z) \\ &+ (N_x - d_x)(N_y + 1 - d_y)(N_z - d_z) \\ &+ (N_x + 1 - d_x)(N_y - d_y)(N_z - d_z)]. \end{aligned} \tag{6}$$

Examination of (6) reveals the four interlocked simple cubic lattices and their altered extents. Furthermore, following an approach similar to that of the b.c.c. internal separations, it can be shown that for the same arbitrary cell as Fig. 1 (with only the base cubic lattice shown) the cross terms can be evaluated using the formulae below. Owing to the nature of the f.c.c. lattice, these distances are only valid when a given pair of the arbitrary cell extents are non-zero. These formulae represent the distances from the base cell atom to all atoms within the three additional lattices and between the additional lattices.

For $d_x \neq 0$ and $d_y \neq 0$,

$$r(d_x, d_y, d_z) = [(d_x - \tfrac{1}{2})^2 + (d_y - \tfrac{1}{2})^2 + d_z^2]^{1/2} \tag{7}$$

$$M(d_x, d_y, d_z) = 2^{\alpha+1}(N_x + 1 - d_x)(N_y + 1 - d_y)(2N_z + 1 - 2d_z) \tag{8}$$

and likewise for $d_x \neq 0, d_z \neq 0$ and $d_y \neq 0, d_z \neq 0$.

A set of routines were written in the MATLAB matrix programming language to calculate the multiplicity function using both the traditional approach (Marciniak *et al.*, 1996) and the new algorithm. These routines were used to generate the powder diffraction profile for cubic crystallites with extent 3 × 3 × 3 in each of the cubic, body-centred cubic and face-centred cubic structures, and in addition a f.c.c. lattice of size 50 × 50 × 50 for a more realistic pattern. The results of these calculations are shown in Fig. 2. The patterns show a formal divergence for small-angle scattering and this part of the curve, though predicted accurately by this method under the first Born approximation, is omitted for convenience.

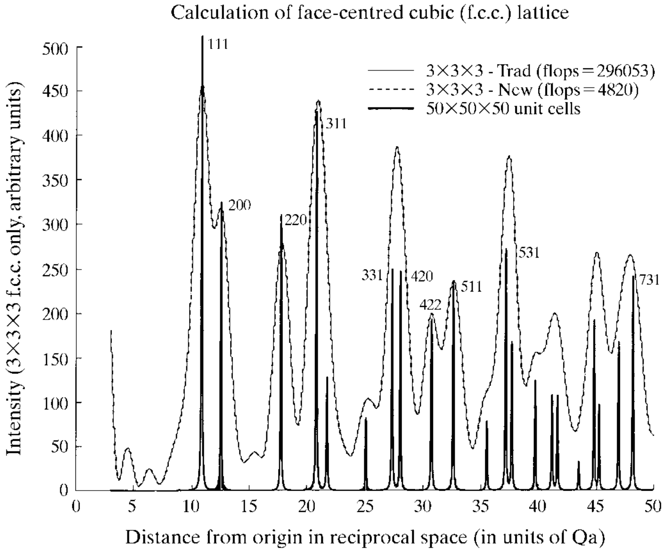

Figure 2
Results of traditional and new calculations using the Debye equation for a face-centred cubic lattice of extent 3 × 3 × 3 unit cells. Also shown is the result of the calculation for a more realistic crystallite of size 50 × 50 × 50 unit cells, where several of the major lines are identified. The small-angle scattering has been omitted for convenience.

Table 1
The number of floating-point operations required to determine $M(r_{ij})$ for various numbers of unit cells in each crystallite.

$N_x \times N_y \times N_z$	Traditional	New
Cubic 1 × 1 × 1	637	145
Cubic 3 × 3 × 3	40942	1118
B.c.c. 1 × 1 × 1	812	219
B.c.c. 3 × 3 × 3	82877	1918
F.c.c. 1 × 1 × 1	1976	443
F.c.c. 3 × 3 × 3	296053	4820

Of central interest here is the improved efficiency of the calculations. Table 1 shows the number of floating-point operations ($N_{\rm flops}$) required to perform the evaluation of the multiplicity function. It can be readily verified that the traditional approach scales as reported in Marciniak *et al.* (1996) with the addition of the self-terms. The resulting relationship has the form $N_{\rm flops} \approx 10 \times n^2$, where the crystals contain 8, 64, 9, 91, 14 and 172 atoms, respectively, and where the factor of ten is a result of the specific MATLAB code used. Two features are apparent in the new approach: firstly, the improved computational efficiency as a result of the different scaling relation with N; and, secondly, the minor increases in $N_{\rm flops}$ as the structure type changes. The latter can be associated directly with the use of an additional evaluation per calculation cell for the b.c.c. case and three additional calculations per calculation cell for the f.c.c. case, corresponding to the interlattice distances. Now, with ten cells along the side of a simple cubic crystal, the new approach requires 22812 operations whereas the traditional approach requires approximately 17.7 million. Furthermore, the scaling is consistent with the anticipated N^3, where N is the number of cells along a side.

We expect the computational advantage of the new approach may eventually yield a significant improvement in the field of powder identification. Although the solution presented in this paper considers only cubic crystallite structures, this method can be extended to encompass triclinic lattices in a straightforward manner. A solution for and ellipsoidal crystallite of general shape is currently being sought. The incorporation of these two features within this approach is both achievable and significant.

Further improvements can be made through noting that each step of the solution is independent of all others, allowing parallel processing to be implemented. It is likely that a *real-time* powder fitting method can be developed in which the data can be fitted during or immediately following acquisition. This method, like the Rietveld method, utilizes the entire powder diffraction curve and therefore allows a significantly higher level of certainty in an identification.

The utilization of algebraic techniques to develop a new algorithm for the calculation of the distance multiplicity function allows substantial improvement in computational efficiency over the traditional approach. Indeed, this approach provides such a computational advantage that the development of a real-time powder pattern fitting system appears possible.

References

Debye, P. (1915). *Ann. Phys. (Leipzig)*, **46**, 809–823.

Marciniak, H., Gierlotka, S. & Palosz, B. (1996). *Mater. Sci. Forum*, **228–231**, 39–42.

Rietveld, H. M. (1967). *Acta Cryst.* **22**, 151–152.

Warren, B. F. (1990). *X-ray Diffraction*, p. 116. Reprint: originally published by Addison-Wesley, Reading, MA, 1969.