

The K -level widths increase rapidly with atomic number beyond iodine, up to 96 eV for uranium. It seems unlikely that K -edge resonances for any of these heavy elements will be strong enough to compensate for the spreading of effects over wavelength any better than in the present case. No L absorption

edges occur in existing elements above 33 keV. Thus the high-energy region is not favorable for applications which would exploit this kind of dichroism or birefringence. Those who seek to avoid these complications may welcome this result. Another conclusion is that the search for examples of greatest anisotropy should be directed toward the long-wavelength region where levels are narrow.

We thank the staff members of CHESS whose cooperation and advice allowed us to complete this experiment with unfamiliar equipment in a single visit; we are particularly indebted to Finn Nielsen, Wilfried Schildkamp and Don Bilderback for assistance and to Philip Coppens and Boris Batterman for helpful discussions. This research was supported by the National Science Foundation under Grants CHE-8217443 and CHE-8515298. The measurements were made at CHESS which is supported by the National Science Foundation. Some facilities of the Lawrence Berkeley Laboratory, supported by the Director, Office of Energy Research, Office of Basic Energy Sciences, Chemical Sciences Division of the US Department of Energy under Contract No. DE-AC03-76SF00098 were used for analysis of the results.

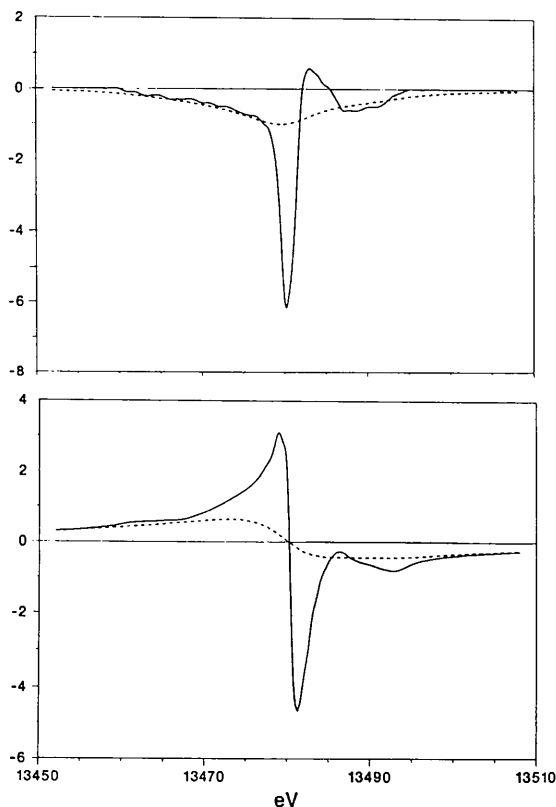


Fig. 3. Polarization anisotropy of anomalous scattering in sodium bromate, $f''_{\sigma} - f''_{\pi}$ above, $f'_{\sigma} - f'_{\pi}$ below. Solid curves show the actual values, broken curves the result of convolution with a line shape of 10.6 eV FWHM.

References

- BEARDEN, J. A. (1967). *Rev. Mod. Phys.* **39**, 78-124.
 BOER, J. L. DE, VAN BOLHUIS, F., OLTHOF-HAZEKAMP, R. & VOS, A. (1966). *Acta Cryst.* **21**, 841-843.
 KRAUSE, M. O. & OLIVER, J. H. (1979). *J. Phys. Chem. Ref. Data*, **8**, 329-338.
 NIELSEN, F. S., LEE, P. & COPPENS, P. (1986). *Acta Cryst.* **B42**, 359-364.
 ROSENZWEIG, A. & MOROSIN, B. (1966). *Acta Cryst.* **20**, 758-761.
 SVENSSON, C., ALBERTSSON, J., LIMINGA, R., KVICK, Å. & ABRAHAMS, S. C. (1983). *J. Chem. Phys.* **78**, 7343-7352.
 TEMPLETON, D. H. & TEMPLETON, L. K. (1985a). *Acta Cryst.* **A41**, 133-142.
 TEMPLETON, D. H. & TEMPLETON, L. K. (1985b). *Acta Cryst.* **A41**, 365-371.

Acta Cryst. (1989). **A45**, 42-50

A Memory-Efficient Fast Fourier Transformation Algorithm for Crystallographic Refinement on Supercomputers

BY AXEL T. BRÜNGER

*The Howard Hughes Medical Institute and
 Department of Molecular Biophysics and Biochemistry, Yale University, New Haven, CT 06511, USA*

(Received 19 April 1988; accepted 20 July 1988)

Abstract

A vectorizable algorithm for fast computation of structure factors and derivatives during refinement of macromolecular structures is presented. It is based

on fast Fourier transformations on subgrids that cover the unit cell of the crystal. The use of subgrids allows reduction of the total memory requirements for the computations without producing large overheads. The algorithm is applicable to all space groups. The

performance of the algorithm on a conventional scalar computer as well as on supercomputers is discussed.

Introduction

The step in crystallographic refinement that consumes the most central processor unit (CPU) time is the evaluation of the crystallographic residual and its first derivatives with respect to the atomic coordinates of the molecule. This applies to restrained least-squares refinement (Sussmann, Holbrook, Church & Kim, 1977; Jack & Levitt, 1978; Konnert & Hendrickson, 1980; Moss & Morffew, 1982) as well as to refinement by simulated annealing (Brünger, Kuriyan & Karplus, 1987; Brünger, Karplus & Petsko, 1989; Brünger, 1988*a, b*). The CPU time needed to compute the structure factors through the direct summation method exceeds the CPU time for a single evaluation of the stereochemical restraints by up to two orders of magnitude. For example, in the case of aspartate aminotransferase, a protein composed of 396 amino acids, the evaluation of the crystallographic residual and its first derivatives through direct summation takes 103 s at 2.8 Å resolution whereas the evaluation of the stereochemical restraints takes only 2.8 s on a Cray-XMP. The efficient calculation of the crystallographic residual and its first derivatives is of particular concern for crystallographic refinement by simulated annealing since this method requires many more updates of the structure factors than restrained least-squares refinement.

Evaluation of the atomic electron density on a finite grid followed by fast Fourier transformation (FFT) provides a way to improve significantly the speed of the calculation of the crystallographic residual and its first derivatives (Cooley & Tukey, 1965; Ten Eyck, 1973, 1977; Agarwal, 1978, 1981; Isaacs, 1984; Tronrud, Ten Eyck & Matthews, 1987; Finzel, 1987). Lunin & Urzhumtsev (1985) showed that this can be generalized to an arbitrary target function involving the observed and computed structure factors by using a multidimensional version of the chain rule in complex space.

This work was motivated by the fact that the presently available FFT programs for crystallographic refinement use external disks to store the grid points of the atomic electron density. The input/output operations associated with such external disk storage decrease the performance of the program on a supercomputer unless all electron density grid points can be fitted into the memory of the computer. At present there are many supercomputers in operation without large memory or special solid-state devices. Therefore, the goal of this work was to develop a space-group-general algorithm that would be easy to implement and adaptable to the available resources of the particular machine. The FFT refinement method of

Agarwal (1981) was reformulated in order to achieve vectorization on supercomputers and minimize total memory requirements. The availability of optimized library routines on supercomputers that evaluate three-dimensional FFT's greatly improved the performance of the algorithm.

The two main differences with earlier FFT implementations (Agarwal, 1981; Tronrud, Ten Eyck & Matthews, 1987) concern the organization of the finite grid and the application of symmetry operators in reciprocal space. If the electron density grid is too large to fit into the available memory of the computer, the FFT is carried out on subgrids and the results are then accumulated. The subgrid can be chosen small enough to fit into the memory of the computer, thus optimizing the performance of the FFT.

It has already been pointed out (Isaacs, 1984) that the modelling of the electron density on the finite grid is an expensive part of the computation. To reduce the time spent in the calculation of the atomic electron density, it is computed only for a set of unique atoms. The FFT is carried out over the unit cell, and the transposed symmetry operators of the crystal space group are then applied to the computed structure factors. This yields the identical result to computing the electron density for all atoms in the unit cell.

The following sections describe the mathematical basis for the subgrid fast Fourier transformation (SGFFT) algorithm and its current implementation on supercomputers. In the last section the performance of the algorithm is discussed for three representative examples.

Notation

We define Ω as a unique set of observed reflections with indices hkl that satisfy certain selection criteria. Usually, the reflections are selected based on their significance, *i.e.* whether the intensity is larger than a certain multiple of the standard deviation. The index triple (h, k, l) is denoted as \mathbf{h} . The space group of the crystal will be described in terms of n_s symmetry operators $(\mathcal{O}_s, \mathbf{t}_s)$ composed of the 3×3 matrix \mathcal{O}_s representing a rotation and a vector \mathbf{t}_s representing a translation. It is assumed that the atomic model consists of n_a unique (not symmetry-related) atoms. The set of unique atoms may coincide with a standard crystallographic asymmetric unit, but this is not a necessary condition for this work. The coordinates of every atom in the unit cell can then be generated by application of a symmetry operator to the coordinates of an atom in the set of unique atoms. The vector \mathbf{r}_i denotes the orthogonal coordinates of atom i in Å. \mathcal{F} is the 3×3 matrix that converts orthogonal Å coordinates into fractional coordinates; \mathcal{F}^* denotes the transpose of it. The columns of \mathcal{F}^* are equal to the reciprocal unit-cell vectors $\mathbf{a}, \mathbf{b}, \mathbf{c}$. The length of

the reciprocal-lattice vector $\mathbf{s} = \mathcal{F}^* \mathbf{h}$ which corresponds to the reflection \mathbf{h} is equal to $2 \sin \theta / \lambda$, where λ is the wavelength of the X-rays and θ is the scattering angle.

Target function

Crystallographic refinement can be understood as a nonlinear optimization problem with the aim of finding a minimum close to the global minimum of a target function T ,

$$\begin{aligned} T(p_1, p_2, \dots, p_n) \\ = \Delta[|F_{\text{obs}}|, \varphi_{\text{obs}}, F_{\text{calc}}(p_1, p_2, \dots, p_n)] \\ + E(p_1, p_2, \dots, p_n) \end{aligned} \quad (1)$$

where Δ is a function of the observed structure-factor amplitudes ($|F_{\text{obs}}|$), observed phase information (φ_{obs}), and the calculated structure factors of the atomic model (F_{calc}). $E(p_1, p_2, \dots, p_n)$ comprises stereochemical and other interactions of the macromolecule; the quantities p_1, p_2, \dots, p_n represent the variables of the system, such as the atomic positions or individual atomic B factors. The FFT refinement programs of Agarwal (1978) and Tronrud, Ten Eyck & Matthews (1987) use steepest descent or conjugate gradient minimization (Fletcher & Reeves, 1964) to optimize T . The recently developed method of crystallographic refinement by simulated annealing (Brünger, Kuriyan & Karplus, 1987; Brünger, Karplus & Petsko, 1989; Brünger, 1988a) uses molecular dynamics to overcome local minima. Molecular dynamics as well as conjugate gradient minimization require the repeated computation of T and its first derivatives with respect to the model parameters p_i . Higher-order minimization methods require the computation of second derivatives as well. Since the computation of $E(p_1, p_2, \dots, p_n)$ and its first derivatives is straightforward (*cf.* Konnert & Hendrickson, 1980) we are only concerned here with the computation of Δ and its first derivatives.

In crystallographic refinement the function Δ usually consists of the crystallographic residual

$$\sum_{\mathbf{h} \in \Omega} W_{\mathbf{h}} [|F_{\text{obs}}(\mathbf{h})| - k |F_{\text{calc}}(\mathbf{h})|]^2, \quad (2)$$

which is a sum of the weighted differences between observed ($|F_{\text{obs}}|$) and calculated ($|F_{\text{calc}}|$) structure-factor amplitudes. The sum extends over the set of all observed reflections Ω . $W_{\mathbf{h}}$ are individual weights for each reflection \mathbf{h} . The scale factor k is set to the value which makes the derivative of the crystallographic residual with respect to k zero,

$$k = \sum_{\mathbf{h} \in \Omega} W_{\mathbf{h}} |F_{\text{obs}}(\mathbf{h})| |F_{\text{calc}}(\mathbf{h})| \left/ \left[\sum_{\mathbf{h} \in \Omega} W_{\mathbf{h}} |F_{\text{calc}}(\mathbf{h})|^2 \right] \right. \quad (3)$$

This is a necessary condition to make the crystallographic residual minimal.

It should be noted that the following derivations do not depend on the particular form of $\Delta(|F_{\text{obs}}|, \varphi_{\text{obs}}, F_{\text{calc}})$. For example, phase information can be added to the crystallographic residual (*cf.* Brünger, 1988b).

Calculated structure factors

The structure factors of a crystal are given by Fourier transformation of the electron density,

$$F_{\text{calc}}(\mathbf{h}) = \int_{\mathbf{g} \in \mathcal{V}} \rho(\mathcal{F}^{-1} \mathbf{g}) \exp(2\pi i \mathbf{h} \cdot \mathbf{g}) \, d\mathbf{g}. \quad (4)$$

The vectors \mathbf{g} are defined in fractional coordinate space. The integration is carried out over the standard unit cell $\mathcal{V} = (0 \dots 1, 0 \dots 1, 0 \dots 1)$. The argument $\mathbf{r} = \mathcal{F}^{-1} \mathbf{g}$ of the electron density ρ is specified in orthogonal Å coordinates.

The electron density ρ is a superposition of the individual electron densities of all atoms in the unit cell of the crystal. Using the crystallographic symmetry operators one can write

$$\rho(\mathbf{r}) = \sum_{s=1}^{n_s} \rho^A(\mathcal{O}_s \mathbf{r} + \mathbf{t}_s) \quad (5)$$

where the sum runs through all symmetry operators ($\mathcal{O}_s, \mathbf{t}_s$) of the space group. The term ρ^A is the superposition of the electron densities ρ_i of a set of unique (*i.e.* not symmetry-related) atoms,

$$\rho^A(\mathbf{r}) = \sum_{i=1}^{n_a} \rho_i(\mathbf{r}) \quad (6)$$

where n_a is the number of unique atoms. Let $F_{\text{calc}}^A(\mathbf{h})$ be the structure factors generated by this unique set of atoms,

$$F_{\text{calc}}^A(\mathbf{h}) = \int_{\mathbf{g} \in \mathcal{V}} \rho^A(\mathcal{F}^{-1} \mathbf{g}) \exp(2\pi i \mathbf{h} \cdot \mathbf{g}) \, d\mathbf{g}. \quad (7)$$

Using (5) and (4) one can derive the result

$$F_{\text{calc}}(\mathbf{h}) = \sum_{s=1}^{n_s} F_{\text{calc}}^A(\mathcal{O}_s^* \mathbf{h}) \exp(2\pi i \mathbf{h} \cdot \mathbf{t}_s) \quad (8)$$

where \mathcal{O}_s^* denotes the transpose of \mathcal{O}_s . The coefficient $\exp(2\pi i \mathbf{h} \cdot \mathbf{t}_s)$ arises from the translational part of the symmetry operator ($\mathcal{O}_s, \mathbf{t}_s$). Thus, the structure factors $F_{\text{calc}}(\mathbf{h})$ can be obtained by computing the structure factors of a unique set of atoms and then accumulating all symmetry-related structure factors.

One possibility for explicit computation of (4) is to carry out the Fourier transformation for each atom translated to the origin, which yields the familiar 'direct summation' formula,

$$F_{\text{calc}}(\mathbf{h}) = \sum_{i=1}^{n_a} f_i(\mathbf{h}) \sum_{s=1}^{n_s} \exp[2\pi i \mathbf{h} \cdot (\mathcal{O}_s \mathcal{F} \mathbf{r}_i + \mathbf{t}_s)]. \quad (9)$$

This equation is valid for all space groups. More efficient expressions can be obtained for each space

group separately by appropriate factoring (*International Tables for X-ray Crystallography*, 1952). This method is used, for instance, in the refinement program *PROLSQ* (Konnert & Hendrickson, 1980).

The form factors $f_i(\mathbf{h})$ are Fourier transforms of the electron densities ρ_i of atom i shifted to the origin. In most cases, the atomic form factors $f_i(\mathbf{h})$ can be approximated by an expression consisting of several Gaussians, such as

$$f_i(\mathbf{h}) = \left\{ \sum_{k=1}^4 a_{ki} \exp[-b_{ki}(\mathcal{F}^*\mathbf{h})^2/4] + a_{0i} \right\} \times \exp[-B_i(\mathcal{F}^*\mathbf{h})^2/4]. \quad (10)$$

The constant B_i is the isotropic B factor of atom i . The constants a_{ki} and b_{ki} can be obtained from *International Tables for X-ray Crystallography* (1974). The following derivations do not depend on the particular shape of the atomic form factors. In particular, the conclusions are valid for anisotropic form factors.

First derivatives of $F_{\text{calc}}(\mathbf{h})$ with respect to atomic positions can be derived from (9),

$$\partial F_{\text{calc}}(\mathbf{h})/\partial r_i = 2\pi i f_i(\mathbf{h}) \sum_{s=1}^{n_s} \mathbf{h} \cdot \mathcal{O}_s \mathcal{F} \mathbf{1} \times \exp[2\pi i \mathbf{h} \cdot (\mathcal{O}_s \mathcal{F} \mathbf{r}_i + \mathbf{t}_s)] \quad (11)$$

where $\mathbf{1}$ is the identity vector.

Fast Fourier transformation

The computation of the exponential coefficients that occur in the direct summation formula [(9)] is expensive. An alternative to the direct summation method is to approximate (4) by discrete Fourier transformation of the electron density sampled at discrete grid points (Ten Eyck, 1977). A three-dimensional grid that covers the unit cell of the crystal is given by a set (Γ) of three-dimensional points

$$\Gamma = \{(a/N_a, b/N_b, c/N_c); \\ a = 0, \dots, N_a - 1; b = 0, \dots, N_b - 1; \\ c = 0, \dots, N_c - 1\}. \quad (12)$$

The points ($\mathbf{g} \in \Gamma$) are specified in fractional coordinates. The grid Γ contains $N_a N_b N_c$ points.

Let the discrete Fourier transformation of $\rho(\mathbf{r})$ over the grid Γ be defined as

$$\text{FT}(\rho, \Gamma; \mathbf{h}) = \sum_{\mathbf{g} \in \Gamma} \rho(\mathcal{F}^{-1}\mathbf{g}) \exp(2\pi i \mathbf{g} \cdot \mathbf{h}) \quad (13)$$

where the sum ($\sum_{\mathbf{g} \in \Gamma}$) represents a three-dimensional sum over all grid points, *i.e.*

$$\sum_{\mathbf{g} \in \Gamma} = \sum_{a=0}^{N_a-1} \sum_{b=0}^{N_b-1} \sum_{c=0}^{N_c-1}. \quad (14)$$

Substitution of the continuous Fourier transformation in (4) by a discrete Fourier transformation over

the finite grid Γ yields an approximation for the structure factors,

$$F_{\text{calc}}(\mathbf{h}) \approx \mathcal{NFT}(\rho, \Gamma; \mathbf{h}) \quad (15)$$

where \mathcal{N} is a normalization constant given by $|\mathbf{a}||\mathbf{b}||\mathbf{c}|/(N_a N_b N_c)$. According to the Nyquist theorem (Brigham, 1974), (15) becomes exact when the electron density ρ can be represented by a Fourier summation with coefficients not larger than $N_a/2$, $N_b/2$, $N_c/2$ in each dimension. Since the Gaussian form factors in (10) have an infinite Fourier spectrum, this implies that the accuracy of (15) is the better the finer the grid Γ chosen. However, a very fine grid decreases the efficiency of the FFT method. At a given grid size the accuracy of (15) can be improved by artificially increasing the B factor for each atom by a constant amount B_0 which is eliminated after Fourier transformation (Cochran, 1948; Ten Eyck, 1977). This broadens the electron density distribution and reduces the influence of the higher-order Fourier coefficients. For the choice of the coarseness of the grid and the artificial B -factor increase B_0 , see Ten Eyck (1977). Typically, B_0 is set to $20 \cdot 0 \text{ \AA}^2$ and the grid is chosen approximately one third of the high-resolution limit of the data.

In principle, the electron density would have to be computed at all grid points specified in Γ . In the special case of the multiple Gaussian approximation (10), the electron density ρ_i for atom i is given by

$$\rho_i(\mathbf{r}) = \sum_{k=1}^4 a_{ki} \left(\frac{4\pi}{b_{ki} + B_i + B_0} \right)^{3/2} \exp \left[\frac{-4\pi^2(\mathbf{r} - \mathbf{r}_i)^2}{b_{ki} + B_i + B_0} \right] \quad (16)$$

which follows by Fourier transformation of the form factors f_i . Thus, this expression would have to be computed at each grid point and for each atom. This would actually require more calculations than the direct summation method. However, since the electron density of individual atoms falls off rapidly, it is a good approximation to compute only $\rho_i(\mathbf{r})$ in the neighborhood around the atoms, *i.e.*

$$\rho_i(\mathbf{r}) \approx \widehat{\rho}_i(\mathbf{r}) = \begin{cases} \rho_i(\mathbf{r}) & \text{for } \mathcal{F}\mathbf{r} \in \Lambda_i \\ 0 & \text{otherwise,} \end{cases} \quad (17)$$

where the neighborhood Λ_i is defined as the set of grid points \mathbf{g} for which $\rho(\mathcal{F}^{-1}\mathbf{g})$ is greater than 10^{-7} \AA^{-3} . This results in spherical neighborhoods with radii of typically $3 \cdot 0$ – $6 \cdot 0 \text{ \AA}$ depending on the atom type and B factor. The artificial increase in the temperature factor (B_0) makes the neighborhoods of the atoms somewhat larger. The definition of Λ_i is complicated by the fact that the neighborhood of an atom can penetrate the unit-cell boundaries. In this case, the selected grid points have to be projected back into the primary unit cell.

From (17) one obtains an approximation for ρ^A ,

$$\widehat{\rho^A(\mathbf{r})} = \sum_{i=1}^{n_a} \widehat{\rho_i(\mathbf{r})}. \quad (18)$$

With (13) and (8) this yields an approximate expression for the structure factors,

$$\widehat{F_{\text{calc}}(\mathbf{h})} = \mathcal{N} \exp[B_0(\mathcal{F}^*\mathbf{h})^2/4] \sum_{s=1}^{n_s} \exp(2\pi i\mathbf{h}\cdot\mathbf{t}_s) \times \text{FT}(\widehat{\rho^A}, \Gamma; \mathcal{O}_s^*\mathbf{h}). \quad (19)$$

The factor in front of the sum eliminates the artificial B -factor increase in (16). Inserting (19) into (1) one obtains an approximation for Δ :

$$\hat{\Delta} = \Delta(|F_{\text{obs}}|, \varphi_{\text{obs}}, \widehat{F_{\text{calc}}}). \quad (20)$$

Since the symmetry operators are applied after Fourier transformation, the electron density grid has to be computed only for the set of unique atoms. An alternative method would be to apply the symmetry operators in orthogonal coordinate space to ρ^A on each grid point that has non-zero density.

Factoring

The Fourier transformation in (19) can be obtained most efficiently by fast Fourier transformation (FFT) (Cooley & Tukey, 1965). The FFT memory requirement is of the order $N_a N_b N_c$. It is shown below that (19) can be factorized, thus reducing the memory requirements. This can be accomplished along similar lines to the reduction of a one-dimensional Fourier transformation into two one-dimensional Fourier transformations of half the size, which provides the basis for the one-dimensional FFT algorithm (Cooley & Tukey, 1965). The use of subgrids was also suggested by Raftery, Sawyer & Pawley (1985) in order to reduce memory requirements for their one-dimensional FFT algorithm to compute structure factors. It is shown below that the use of subgrids can be generalized to three dimensions.

A three-dimensional subgrid is defined by a set of points

$$\Gamma'' = \{(a''/N_a'', b''/N_b'', c''/N_c''); \\ a'' = 0, \dots, N_a'' - 1; b'' = 0, \dots, N_b'' - 1; \\ c'' = 0, \dots, N_c'' - 1\} \quad (21)$$

where $N_a'', N_b'', N_c'', N'_a, N'_b, N'_c$ are integer numbers such that

$$\begin{aligned} N'_a N''_a &= N_a, & N'_b N''_b &= N_b, & N'_c N''_c &= N_c, \\ N'_a < N_a, & N'_b < N_b, & N'_c < N_c, & (22) \\ N''_a \leq N_a, & N''_b \leq N_b, & N''_c \leq N_c. \end{aligned}$$

The full grid Γ can be generated by combining all

points of Γ'' and a set of points Γ' defined by

$$\begin{aligned} \Gamma' &= \{(a'/N_a, b'/N_b, c'/N_c); \\ a' &= 0, \dots, N'_a - 1; b' = 0, \dots, N'_b - 1; \\ c' &= 0, \dots, N'_c - 1\}. \end{aligned} \quad (23)$$

The points in Γ' provide the offset that is added to the subgrid Γ'' . We can then formally write

$$\Gamma = \Gamma' + \Gamma'' \quad (24)$$

where the '+' operator adds all points of set Γ' to all points of set Γ'' . Fig. 1 illustrates the factoring of Γ for a special two-dimensional case.

With these definitions one can rewrite (13),

$$\begin{aligned} \text{FT}(\rho, \Gamma; \mathbf{h}) &= \sum_{\mathbf{g}' \in \Gamma'} \sum_{\mathbf{g}'' \in \Gamma''} \exp[2\pi i(\mathbf{g}' + \mathbf{g}'') \cdot \mathbf{h}] \\ &\times \rho[\mathcal{F}^{-1}(\mathbf{g}' + \mathbf{g}'')]. \end{aligned} \quad (25)$$

By making use of the relation $\exp(a+b) = \exp(a)\exp(b)$ one may factorize this equation, *i.e.*

$$\begin{aligned} \text{FT}(\rho, \Gamma; \mathbf{h}) &= \sum_{\mathbf{g}' \in \Gamma'} \exp(2\pi i\mathbf{g}' \cdot \mathbf{h}) \text{FT}(\rho, \Gamma'' + \mathbf{g}'; \mathbf{h} \bmod \Gamma'') \end{aligned} \quad (26)$$

where the Fourier transformation of ρ on the right-hand side is given by

$$\begin{aligned} \text{FT}(\rho, \Gamma'' + \mathbf{g}'; \mathbf{h}) &= \sum_{\mathbf{g}'' \in \Gamma''} \exp(2\pi i\mathbf{g}'' \cdot \mathbf{h}) \rho[\mathcal{F}^{-1}(\mathbf{g}' + \mathbf{g}'')]. \end{aligned} \quad (27)$$

The use of $\mathbf{h} \bmod \Gamma''$ rather than \mathbf{h} in (26) reflects the fact that the projection of the reflection indices into the periodic subgrid Γ'' does not change the value of $\exp(2\pi i\mathbf{g}'' \cdot \mathbf{h})$. The mod function operates on each component h, k, l of the vector \mathbf{h} separately, *i.e.* we define

$$\mathbf{h} \bmod \Gamma'' = \begin{cases} h \bmod N''_a \\ k \bmod N''_b \\ l \bmod N''_c \end{cases} \quad (28)$$

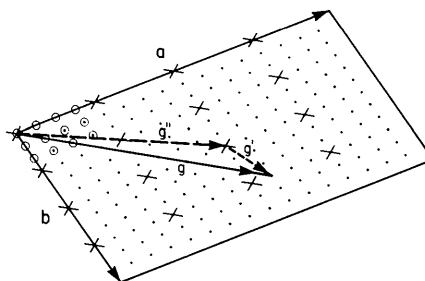


Fig. 1. Illustration of the factoring of the grid Γ (small dots) into the subgrids Γ'' (crosses) and Γ' (circles) for a two-dimensional case. The vectors \mathbf{a} and \mathbf{b} are unit-cell vectors. The grid point \mathbf{g} can be represented as $\mathbf{g}' + \mathbf{g}''$. In this example, the Fourier transformation over the grid Γ is a sum of 12 Fourier transformations over the subgrid Γ'' where each subgrid is translated by a vector $\mathbf{g}' \in \Gamma'$.

where $x \bmod y$ is defined as the remainder of the integer division of x by y . This projection of the reflection indices \mathbf{h} into the subgrid Γ'' is necessary since the result of the subgrid Fourier transform [(27)] will only be available for $\mathbf{h} \in \Gamma''$. By inserting (26) into (19) we obtain for the calculated structure factors the expression

$$\begin{aligned} \widehat{F}_{\text{calc}}(\mathbf{h}) &= \mathcal{N} \exp[B_0(\mathcal{F}^* \mathbf{h})^2/4] \\ &\times \sum_{\mathbf{g}' \in \Gamma'} \sum_{s=1}^{n_s} \exp[2\pi i(\mathbf{g}' \cdot \mathcal{O}_s^* \cdot \mathbf{h} + \mathbf{t}_s \cdot \mathbf{h})] \\ &\times \text{FT}(\widehat{\rho}^A, \Gamma'' + \mathbf{g}'; \mathcal{O}_s^* \mathbf{h} \bmod \Gamma''). \end{aligned} \quad (29)$$

The summation over Γ' has to be carried out for the number $|\Omega|$ of the set of observed reflections Ω . This requires memory of the order of $|\Omega|$. The Fourier transformations of $\widehat{\rho}^A$ require memory of the order of $N_a'' N_b'' N_c''$. Thus, the total memory requirement for (29) is of the order of $N_a'' N_b'' N_c'' + |\Omega|$. The potential drawback of (29) appears to be the computation of $N_a'' N_b'' N_c'' n_s$ exponential coefficients. Unless additional memory is used to precompute and store these coefficients, they have to be recomputed each time the structure factors are required. Equation (29) provides the basis of the SGFFT algorithm.

First derivatives

In this section we will show how to evaluate the first derivatives of $\widehat{\Delta}$ [(20)] with respect to all model parameters p_i . As was pointed out by Cochran (1948), the chain rule can be used to compute the derivatives of the crystallographic residual. However, since $\widehat{\Delta}$ is an approximation of Δ , the derivatives of $\widehat{\Delta}$ will then be approximations of the derivatives of Δ . The following derivation generalizes the work of Lunin & Urzhumtsev (1985) to the SGFFT algorithm introduced in the previous section.

Using the chain rule one can write

$$\frac{\partial \widehat{\Delta}}{\partial p_i} = \sum_{\mathbf{h} \in \Omega} \frac{\widehat{\partial \Delta}}{\partial \widehat{F}_{\text{calc}}^R(\mathbf{h})} \frac{\partial \widehat{F}_{\text{calc}}^R(\mathbf{h})}{\partial p_i} + \frac{\widehat{\partial \Delta}}{\partial \widehat{F}_{\text{calc}}^I(\mathbf{h})} \frac{\partial \widehat{F}_{\text{calc}}^I(\mathbf{h})}{\partial p_i} \quad (30)$$

where $\widehat{F}_{\text{calc}}^R(\mathbf{h})$ and $\widehat{F}_{\text{calc}}^I(\mathbf{h})$ denote the real and the imaginary part of $\widehat{F}_{\text{calc}}(\mathbf{h})$, respectively. From (29) it follows that

$$\begin{aligned} \partial \widehat{F}_{\text{calc}}(\mathbf{h}) / \partial p_i &= \mathcal{N} \exp[B_0(\mathcal{F}^* \mathbf{h})^2/4] \\ &\times \sum_{\mathbf{g}' \in \Gamma'} \sum_{s=1}^{n_s} \exp[2\pi i(\mathbf{g}' \cdot \mathcal{O}_s^* \cdot \mathbf{h} + \mathbf{t}_s \cdot \mathbf{h})] \\ &\times \text{FT}(\partial \widehat{\rho}^A / \partial p_i, \Gamma'' + \mathbf{g}'; \mathcal{O}_s^* \mathbf{h} \bmod \Gamma'') \end{aligned} \quad (31)$$

where we define

$$\frac{\partial \widehat{F}_{\text{calc}}(\mathbf{h})}{\partial p_i} = \frac{\partial \widehat{F}_{\text{calc}}^R(\mathbf{h})}{\partial p_i} + i \frac{\partial \widehat{F}_{\text{calc}}^I(\mathbf{h})}{\partial p_i}. \quad (32)$$

In principle one could compute the derivatives by inserting (31) into (30), but this would be impractical since it would require a Fourier transformation for each parameter p_i of each atom i . It is shown in the following that instead of carrying out the Fourier transformation of $\partial \widehat{\rho}^A / \partial p_i$, one can obtain an expression that involves a Fourier transformation of coefficients that are related to $\widehat{\partial \Delta} / \partial \widehat{F}_{\text{calc}}$. Let us define

$$\frac{\widehat{\partial \Delta}}{\partial \widehat{F}_{\text{calc}}(\mathbf{h})} = \frac{\widehat{\partial \Delta}}{\partial \widehat{F}_{\text{calc}}^R(\mathbf{h})} + i \frac{\widehat{\partial \Delta}}{\partial \widehat{F}_{\text{calc}}^I(\mathbf{h})}, \quad (33)$$

i.e. $\widehat{\partial \Delta} / \partial \widehat{F}_{\text{calc}}(\mathbf{h})$ is a complex number whose real part contains the first derivative of $\widehat{\Delta}$ with respect to the real part of the structure factor $\widehat{F}_{\text{calc}}^R(\mathbf{h})$ and whose imaginary part contains the first derivative of $\widehat{\Delta}$ with respect to the imaginary part of the structure factor $\widehat{F}_{\text{calc}}^I(\mathbf{h})$.

If we make the reasonable assumption that $\widehat{\Delta}$ is a real function, it follows from (30) that

$$\frac{\partial \widehat{\Delta}}{\partial p_i} = \text{Re} \left\{ \sum_{\mathbf{h} \in \Omega} \left[\frac{\partial \widehat{\Delta}}{\partial \widehat{F}_{\text{calc}}(\mathbf{h})} \right]^* \frac{\partial \widehat{F}_{\text{calc}}(\mathbf{h})}{\partial p_i} \right\}, \quad (34)$$

where Re denotes the real part of its argument, and * the complex conjugate. The Fourier transformation in (31) involves a sum over all points $\mathbf{g} \in \Gamma''$. The partial derivatives of $\widehat{\rho}^A$ are zero for grid points outside the neighborhood A_i of atom i ,

$$\begin{aligned} \frac{\partial \widehat{\rho}^A}{\partial p_i}(\mathbf{g}'') &= \sum_{j=1}^{n_a} \frac{\partial \widehat{\rho}_j^A}{\partial p_i}(\mathbf{g}'') = \frac{\partial \widehat{\rho}_i^A}{\partial p_i}(\mathbf{g}'') \\ &= \begin{cases} \partial \rho_i(\mathcal{F}^{-1} \mathbf{g}) / \partial p_i & \mathbf{g} \in A_i, \text{ otherwise} \\ 0 \end{cases} \end{aligned} \quad (35)$$

which follows from (6) and (17). Therefore, it is more efficient to carry out the summation over A_i explicitly rather than by FFT's.

We will show below that the sum over \mathbf{h} in (30) can be replaced by a FFT which uses indices \mathbf{h}'' that are related to \mathbf{h} . The Fourier transformation in (31) involves coefficients of the form

$$\exp[2\pi i \mathbf{g}'' \cdot (\mathcal{O}_s^* \mathbf{h} \bmod \Gamma'')] \quad (36)$$

where $\mathbf{g}'' \in \Gamma'' + \mathbf{g}'$. Since the summation over \mathbf{g}'' is carried out explicitly this leaves

$$\mathbf{h}'' = \mathcal{O}_s^* \mathbf{h} \bmod \Gamma'' \quad (37)$$

as a candidate for Fourier summation. Let us define a set of indices for each $\mathbf{h}'' \in \Gamma''$ and each symmetry operator,

$$\text{map}(\mathbf{h}'', \mathcal{O}_s; \Gamma'') = \{\mathbf{h} \in \Omega \text{ and } (\mathcal{O}_s^* \mathbf{h}) \bmod \Gamma'' = \mathbf{h}''\}. \quad (38)$$

where the mod function has been defined in (28). Let $B(\mathbf{h}'')$ be complex numbers that are obtained from the first derivatives of $\widehat{\Delta}$ after summation over the sets $\text{map}(\mathbf{h}'', \mathcal{O}_s; \Gamma'')$ and application of the symmetry

operators,

$$B(\mathbf{h}'') = \sum_{s=1}^{n_s} \sum_{\mathbf{h} \in \text{map}(\mathbf{h}'', \mathcal{O}_s; \Gamma'')} \exp[2\pi i(\mathbf{g}' \cdot \mathcal{O}_s^* \cdot \mathbf{h} + \mathbf{t}_s \cdot \mathbf{h})] \times [\widehat{\partial \Delta} / \partial F_{\text{calc}}(\mathbf{h})]^* \quad (39)$$

for all $\mathbf{h}'' \in \Gamma''$. The expression in square brackets was defined in (33) above, the * means the complex conjugate. Note that the second sum is defined as zero if the set $\text{map}(\mathbf{h}'', \mathcal{O}_s; \Gamma'')$ is empty. Another way of expressing $B(\mathbf{h}'')$ is to generate for each observed reflection \mathbf{h} the set of symmetry-related reflections. The resulting reflection indices are projected into Γ'' . For each projected index the complex conjugate of the first derivative of $\widehat{\Delta}$ with respect to $F_{\text{calc}}(\mathbf{h})$ is then multiplied by the exponential coefficient as given in (39) and the result is accumulated in $B(\mathbf{h}'')$. With these definitions one obtains by resummation of (31) and (34)

$$\frac{\partial \widehat{\Delta}}{\partial p_i} = \text{Re} \left\{ \sum_{\mathbf{g}'' \in \Gamma''} \sum_{\mathbf{g}' \in \Gamma''} \frac{\widehat{\partial \rho^A}[\mathcal{F}^{-1}(\mathbf{g}'' + \mathbf{g}')] }{\partial p_i} \times \text{FT}(B, \Gamma''; \mathbf{g}'') \right\} \quad (40)$$

where the Fourier transform is applied to $B(\mathbf{h}'')$. Equation (40) allows computation of all partial derivatives using the same number of Fourier transformations needed to compute the structure factors [(29)]. The CPU time and memory requirements for (40) are identical to the ones for (29).

Incorporation of Hermitian symmetry into the SGFFT algorithm

Hermitian symmetry of the structure factors [$F_{\text{calc}}(\mathbf{h}) = F_{\text{calc}}(-\mathbf{h})$] can easily be incorporated into (29) and (40). The number of operations required for

the Fourier transformations in (29) and (40) is then reduced by a factor of two. The set Ω can be reduced to one hemisphere, i.e. without restriction the hemisphere defined by $h > 0$. However, the Fourier transformation of B in (40) includes both hemispheres. This implies that (40) has to be multiplied by a factor $\frac{1}{2}$. The right-hand side of (40) will have a zero imaginary part because of the Hermitian symmetry.

Vectorization of the SGFFT algorithm

Fig. 2 shows a flowchart of the SGFFT algorithm described in (29) and (40). All operations are highly vectorizable with the exception of the computation of the electron density grid. The vectorization was accomplished automatically by the Cray and Convex compilers. Machine-dependent programming was not necessary. The electron density calculation is only moderately vectorizable since the use of look-up tables to compute the form factors [(10)] and the organization of the neighborhoods A_i of the atoms produce memory conflicts. Thus, at the present state of the program, the electron density calculation is memory rather than CPU bound. It is important to note that the reduction of Gaussian factors in (10) to three (as is used by Agarwal, 1978) reduces the CPU time required to compute the electron density on a supercomputer by less than 20% (not shown).

Efficient library routines were used for the three-dimensional FFT's. The algorithm in Fig. 2 is also suitable for parallelization, so that the algorithm should perform well on a multiprocessor machine with shared memory. Largely parallel machines would in fact be better suited for the electron density calculation.

The SGFFT algorithm is part of the *X-PLOR* program (Brünger, Karplus & Petsko, 1989; Brünger, 1988a, b). The program *X-PLOR* is available on request from ATB.

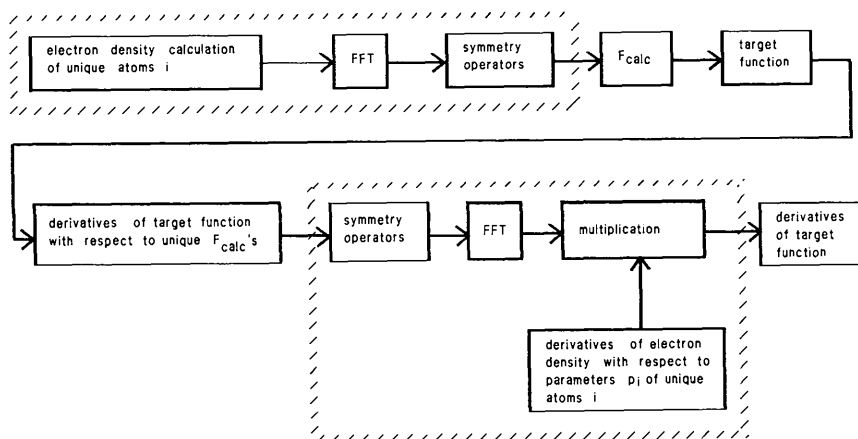


Fig. 2. Flowchart showing the main stages of the algorithm described in this work to compute the structure factors, the target function, and the partial derivatives with respect to the atomic parameters p_i . Blocks enclosed by shaded lines represent (29) and (40) which require large memory.

Table 1. CPU times (s) for computation of the target function Δ and partial derivatives with respect to atomic positions

Task	Vax 8700*	Convex-C1†	Cray-XMP‡
(A) Crambin§			
$P2_1$, 1.5-8.0 Å, 327 non-H atoms, 5576 reflections, $ \Gamma / \Gamma' = 1$			
Direct summation	440.5	72.55	4.23
ρ^A (unique atoms)	11.5	11.52	2.35
P_1 -FFT	15.7	2.24	0.25
Symmetry, factoring	1.1	0.70	0.06
Total	28.3	14.5	2.66
(B) Aspartate aminotransferase¶			
$C222_1$, 2.8-8.0 Å, 3086 non-H atoms, 8124 reflections, $ \Gamma / \Gamma' = 4$			
Direct summation	10 355	1572	103
ρ^A (unique atoms)	56	41.3	2.97
P_1 -FFT	373.2	21.8	1.67
Symmetry, factoring	34	8.1	0.64
Total	463.2	71.2	5.28
(C) Alkaline phosphatase**			
$I222$, 2.4-8.0 Å, 6829 non-H atoms, 38 286 reflections, $ \Gamma / \Gamma' = 6$			
Direct summation	76 920	11 654	686.5
ρ^A (unique atoms)	170	135.4	15.38
P_1 -FFT	1046	63.0	5.9
Symmetry, factoring	207	53.5	4.4
Total	1423	251.9	25.68
<i>PROFFT</i> ††	602	—	—

* With floating point accelerator unit, 16 Mbyte memory, 64-bit precision. Using IMSL (*Library Manual*, Houston: IMSL Inc., 1982) *FFT3D* routine for 3D FFT.

† Using not more than 16 Mbyte memory, 64-bit precision. Using VECCLIB (Convex Computer Corp., 1988) *Z3DFFT* routine for 3D FFT.

‡ Using not more than 16 Mbyte memory, 64-bit precision. Using SCILIB (Cray Research Inc., 1988) simultaneous FFT routines (*CFTAX* and *CFFTMLT*) to compute 3D Fourier transformation.

§ Using an [81, 40, 48] grid and an identical subgrid.

¶ Using a [180, 96, 96] grid and a [180, 96, 24] subgrid. Owing to systematic absences only half the symmetry operators of space group $C222_1$ have to be applied to compute the structure factors ($[x, y, z]$, $[-x, y, \frac{1}{2} - z]$, $[x, -y, \frac{1}{2} + z]$, $[x, -y, -z]$).

** Using a [240, 216, 96] grid and a [240, 216, 16] subgrid. Owing to systematic absences only half the symmetry operators of space group $I222_1$ have to be used to compute the structure factors ($[x, y, z]$, $[-x, -y, z]$, $[x, -y, -z]$, $[-x, y, -z]$).

†† Using the space-group-specific program *PROFFT* (Finzel, 1987) for space group $I222$ in 32-bit precision.

Results

Table 1 compares the direct-summation algorithm and the SGFFT algorithm on various computers for three representative protein structures. The first case is a 1.5 Å resolution structure of crambin (Hendrickson & Teeter, 1981), a protein composed of 46 amino acids. The second case is a 2.8 Å resolution structure of a mutant of aspartate aminotransferase (Smith, Ringe, Finlayson & Kirsch, 1986), a protein composed of 396 amino acids. The third case is a 2.4 Å resolution structure of alkaline phosphatase (Sowadski, Handschuhmacher, Murthy, Foster & Wyckoff, 1985), a dimeric protein where each monomer is composed of 449 amino acids. The reported CPU times are those required for the computation of the function Δ [(2)] and its first derivatives with respect to the atomic positions (r_i). Equation (9) and a corresponding equation for the first derivatives were used for the direct-summation calculations. The total CPU time for the SGFFT calculation is broken down into three parts, the time spent for the calculations

of the electron density grid, the time spent for the FFT calculations, and the time spent for the summations over the n_s symmetry operators and the Γ' grid points. Equations (16), (17) and (35) were used to compute the electron density ρ^A and its first derivatives. The application of symmetry and subgrid operators was carried out according to (29) and (40).

In the case of alkaline phosphatase, the total CPU time required for the SGFFT algorithm on the Vax 8700 (1423 s) is compared with the CPU time required by the non-vectorizable program *PROFFT* (Finzel, 1987) that makes use of a space-group-specific FFT routine for $I222$ in 32-bit precision (602 s). Thus, the SGFFT algorithm shows reasonable performance on a conventional scalar computer despite the fact that the SGFFT algorithm is space-group general and was executed in 64-bit precision. Execution of the SGFFT algorithm in 32-bit precision on the Vax 8700 results in a 30% reduction of CPU time (not shown).

Table 1 clearly shows that the SGFFT algorithm is up to two orders of magnitude faster than the direct summation method. The SGFFT algorithm appears to perform best for the larger systems, aspartate aminotransferase and alkaline phosphatase. On the Vax 8700 computer, the FFT's require most of the total CPU time whereas on both the superminicomputer (Convex-C1) and the supercomputer (Cray-XMP) the electron density calculation becomes the most expensive part of the computation. This clearly shows that, at the present state of the program, the introduction of space-group-specific FFT's could reduce the total CPU time for the SGFFT method by not more than a factor of two.

The application of the factoring and symmetry operators appears to be almost negligible on the Vax 8700 computer. Even in the case of alkaline phosphatase with a subgrid Γ'' a sixth of the size of the original grid Γ , the operations take not more than a fifth of the time spent for the execution of the FFT's. On the supercomputers, this situation is not quite as favorable. This is mainly because the FFT's are library routines that make use of the particular architecture of the machine, whereas the routine that carried out the factoring and symmetry operations was written in Fortran. Nevertheless, the degree of vectorizability for this routine is high; for example, in the case of alkaline phosphatase the routine is 47 times faster on the Cray-XMP than on the Vax 8700. Nonvectorizable code executes approximately six times faster on the Cray-XMP than on the Vax 8700.

The direct summation method [(9)] appears to be highly vectorizable. This is reflected by the CPU times in Table 1; e.g. the CPU time for the direct summation is increased by a factor 100 between the Cray-XMP and the Vax 8700. The SGFFT algorithm is more complicated than the direct summation method. Despite this, it has been shown that it is possible to vectorize the SGFFT algorithm efficiently. The

vectorized SGFFT algorithm is up to 26 times faster than the vectorized direct summation method on the Cray-XMP. Since the SGFFT algorithm is designed in a general and simple way, it is expected that it could be efficiently implemented on supercomputers with new parallel architectures.

The author thanks M. Karplus, J. Kuriyan, G. A. Petsko and W. I. Weis for useful discussions. The work described in this paper was begun while the author was a research associate at Harvard University.

References

- AGARWAL, R. C. (1978). *Acta Cryst.* **A43**, 791-809.
 AGARWAL, R. C. (1981). In *Refinement of Protein Structures*, pp. 24-28. SRC Daresbury Laboratory, Warrington, England.
 BRIGHAM, E. O. (1974). In *The Fast Fourier Transform*. Englewood Cliffs, NJ: Prentice-Hall.
 BRÜNGER, A. T. (1988a). *Crystallographic Refinement by Simulated Annealing*. In *Crystallographic Computing 4: Techniques and New Technologies*, edited by N. W. ISAACS & M. R. TAYLOR. Oxford Univ. Press.
 BRÜNGER, A. T. (1988b). *J. Mol. Biol.* In the press.
 BRÜNGER, A. T., KARPLUS, M. & PETSKO, G. A. (1989). *Acta Cryst.* **A45**, 50-61.
 BRÜNGER, A. T., KURIYAN, K. & KARPLUS, M. (1987). *Science*, **235**, 458-460.
 COCHRAN, W. (1948). *Acta Cryst.* **1**, 138-142.

- COOLEY, J. W. & TUKEY, J. W. (1965). *Math. Comput.* **19**, 297-301.
 FINZEL, B. C. (1987). *J. Appl. Cryst.* **20**, 53-55.
 FLETCHER, R. & REEVES, C. M. (1964). *Comput. J.* **7**, 149-154.
 HENDRICKSON, W. A. & TEETER, M. A. (1981). *Nature (London)*, **290**, 107-112.
International Tables for X-ray Crystallography (1952). Vol. I. Birmingham: Kynoch Press.
International Tables for X-ray Crystallography (1974). Vol. IV. Birmingham, Kynoch Press. (Present distributor Kluwer Academic Publishers, Dordrecht.)
 ISAACS, N. (1984). In *Methods and Applications in Crystallographic Computing*, edited by S. R. HALL & T. ASHIDA, pp. 193-205. Oxford: Clarendon Press.
 JACK, A. & LEVITT, M. (1978). *Acta Cryst.* **A34**, 931-935.
 KONNERT, J. H. & HENDRICKSON, W. A. (1980). *Acta Cryst.* **A36**, 344-349.
 LUNIN, V. Y. & URZHUMTSEV, A. G. (1985). *Acta Cryst.* **A41**, 327-333.
 MOSS, D. S. & MORFFEW, A. J. (1982). *Comput. Chem.* **6**, 1-3.
 RAFTERY, J., SAWYER, L. & PAWLEY, G. S. (1985). *J. Appl. Cryst.* **18**, 424-429.
 SMITH, D. L., RINGE, D., FINLAYSON, W. L. & KIRSCH, J. F. (1986). *J. Mol. Biol.* **191**, 301-302.
 SOWADSKI, J. M., HANDSCHUMACHER, M. D., MURTHY, H. M. K., FOSTER, B. A. & WYCKOFF, H. W. (1985). *J. Mol. Biol.* **186**, 417-433.
 SUSSMAN, J. L., HOLBROOK, S. R., CHURCH, G. M. & KIM, S. H. (1977). *Acta Cryst.* **A33**, 800-804.
 TEN EYCK, L. F. (1973). *Acta Cryst.* **A29**, 183-191.
 TEN EYCK, L. F. (1977). *Acta Cryst.* **A33**, 486-492.
 TRONRUD, D. E., TEN EYCK, L. F. & MATTHEWS, B. W. (1987). *Acta Cryst.* **A43**, 489-500.

Acta Cryst. (1989). **A45**, 50-61

Crystallographic Refinement by Simulated Annealing: Application to Crambin

BY AXEL T. BRÜNGER

The Howard Hughes Medical Institute and Department of Molecular Biophysics and Biochemistry, Yale University, New Haven, CT 06511, USA, and Department of Chemistry, Harvard University, Cambridge, MA 02138, USA

MARTIN KARPLUS

Department of Chemistry, Harvard University, Cambridge, MA 02138, USA

AND GREGORY A. PETSKO

Department of Chemistry, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

(Received 22 April 1988; accepted 20 July 1988)

Abstract

A detailed description of the method of crystallographic refinement by simulated annealing is presented. To test the method, it has been applied to a 1.5 Å resolution X-ray structure of crambin. The dependence of the success of the simulated annealing protocol with respect to the temperature of the heating

stage is discussed. Optimal success is achieved at relatively high temperatures. Regardless of the protocol used, the molecular-dynamics refined structure always yields an improved *R* factor compared with restrained least-squares refinement without manual re-fitting. The differences between the various refined structures and the corresponding electron density maps are discussed.