A Rietveld-Refinement Program RIETAN-94 for Angle-Dispersive X-Ray and Neutron Powder Diffraction

Fujio IZUMI

National Institute for Research in Inorganic Materials

1-1 Namiki, Tsukuba, Ibaraki 305, Japan

E-mail: izumi@nirim.go.jp

(Revised on March 24, 1997)

"When you can measure what you are speaking about and express it in numbers, you know something about it; but when you cannot express it in numbers, your knowledge is of a meagre and unsatisfactory kind; it may be the beginning of knowledge, but you have scarcely in your thoughts advanced to the state of science, whatever the matter may be."

Lord Kelvin

1. Introduction

The Rietveld method (Rietveld, 1967; Rietveld, 1969) is a technique for refining structure parameters (fractional coordinates, isotropic/anisotropic atomic displacement parameters, and occupation factors, magnetic moments, *etc.*) and lattice parameters directly from whole powder diffraction patterns without separating reflections. This method is very useful (i) when single crystals cannot be grown at all, (ii) when only twinned samples can be prepared, and (iii) when physical and/or chemical properties of single-crystal forms differ from those of polycrystalline ones. It was originally devised for analysis of angle-dispersive neutron diffraction data (Rietveld, 1969) and has been extensively applied to intensity data measured at reactor and pulsed neutron sources (Cheetham & Taylor, 1977; Albinati & Wiles, 1982; David, 1988). Since Malmros & Thomas (1977) first applied the Rietveld method to X-ray diffraction data, it has been widely used for structure refinements with conventional X-ray (Post & Bish, 1989) and synchrotron X-ray (Cox, 1992) diffraction data.

The Rietveld method, *per se*, cannot be applied to unknown structures. However, techniques of *ab initio* structure analysis from powder data have now been developing promptly. For example, a program for automatic solution of cyrstal structures by direct methods optimized for powder data was developed by Altomare *et al*. (1994). Rietveld refinement is used as the last process of *ab initio* structure analysis. Rapid advances in high-resolution transmission electron microscopy (HRTEM) has made it easier to observe crystal-structure images. HRTEM images correspond to Fourier maps in X-ray crystal analysis, serving to construct initial structural models for Rietveld refinement. Thus, the importance of the Rietveld method is still increasing more and more. It is now regarded as a fundamental technique of characterizing polycrystalline materials in the field of physics, chemistry, materials science, mineralogy, *etc*.

Variable parameters, $\boldsymbol{x}$, in Rietveld analysis is much more than those in single-crystal structure refinement in spite of less structural information contained in powder diffraction data. Peak-shift, background, profile-shape, preferred-orientation, and lattice parameters must be refined in addition to scale factors and structure parameters in

Rietveld analysis. The number of refinable parameters increases considerably when dealing with samples consisting of two or more phases. However, the addition of variable parameters should not necessarily be regarded as a serious disadvantage in the Rietveld method. Conversely, Rietveld analysis provides us with a variety of information other than structure parameters. That is, precise lattice parameters are determined from elements of metric tensors (Izumi, 1996), isotropic/anisotropic crystallite sizes and microstrains from profile-shape parameters (*cf.* 2.7), and mass fractions of constituent phases from their scale factors (Hill & Howard, 1987; Hill, 1993).

RIETAN is a Rietveld-refinement program developed by the author (Izumi, 1989; Izumi, 1993; Izumi *et al.*, 1987). The simulation of powder diffraction patterns is also possible with it. Two separate programs were coded in FORTRAN for (i) angle-dispersive (constant-wavelength) X-ray and neutron diffraction (Izumi, 1989; Izumi, 1993) and (ii) time-of-flight (TOF) neutron diffraction at the KENS pulsed neutron scattering facility (Izumi *et al.*, 1987). The former software has been distributed to many laboratories, mainly in Japan, and numerous papers have been published so far with it. The most remarkable advantage of RIETAN over other Rietveld-refinement programs is stable and, in most cases, automatic convergence to a minimum of the weighted sum of residuals. Complex and elaborate algorithms are used in nonlinear least-squares procedures for this purpose.

The angle-dispersive version of RIETAN has recently been revised extensively. Many new features have been implemented, including a modified pseudo-Voigt profile-shape function (Thompson, Cox & Hastings, 1987) made asymmetric by applying a multi-term Simpson's rule integration (Howard, 1982), the preferred-orientation function of March and Dollase (March, 1932; Dollase, 1986), a robust background function, a smart input manner (Kim & Izumi, 1994), and several kinds of formats for graphical data. The most significant improvement is the new profile-shape function that is appealing in terms of its soundness based on physics motivation.

Regrettably, this was the first major change in RIETAN. It is the so-called 'superconductivity fever' that delayed the extensive revision of RIETAN. During the period from 1987 to 1993, the author considerably lessened his commitment to RIETAN because he dedicated himself to the neutron powder diffraction studies of crystal and defect structures for high-T_c superconducting oxides (Izumi, 1993a; Izumi & Takayama-Muromachi, 1995). However, such his activity in the field of high-T_c superconductivity has now made the Rietveld method famous in Japan.

The object of this report is to introduce new features of RIETAN-94 briefly and to present a temporary substitute for its user's manual, which is expected to have been finished by the end of December in 1996 (really?). See Izumi (1989) and Izumi (1993) for information about details in RIETAN not covered in this document. This document plus

templates of input files may sometimes not be adequate for actual Rietveld analysis. Please refer to the source program of rietan1.f for getting information about the method of inputting data which are not included in them.

RIETAN-94, its user's manual, *etc*. stored at the anonymous FTP server of NIRIM will be updated now and then. The latest version of RIETAN-94 can be obtained *via* Internet at any time.

2. Model function

This section deals with several new functions contained in the model function (calculated intensity), $f_i(\boldsymbol{x})$, in RIETAN-94. Various equations and algorithms used in Rietveld-refinement programs will be described in detail in a forthcoming book (Izumi, 1996).

2.1 Profile-shape function

The model function in RIETAN was extensively modified to obtain better fits between observed and calculated patterns and to refine physically meaningful profile-shape and preferred-orientation parameters. In particular, the profile-shape function has been changed from an empirical formula (Izumi, 1989) into a better one as described below. The new profile-shape function gives somewhat lower R factors than the previous one does unless the least-squares solution diverges or falls into a trap of a false minimum.

Symmetric profile-shape function

Pseudo-Voigt (Wertheim *et al.*, 1974) and Pearson VII (Hall *et al.*, 1977) functions have been usually adopted as symmetric profile-shape functions for angle-dispersive X-ray and neutron diffraction (Young & Wiles, 1982). Both functions fit powder diffraction data equally well. Nevertheless, the pseudo-Voigt function is preferred to the Pearson VII function for Rietveld analysis because it can offer physical insight into the origin of the profile shape, *e.g.*, profile broadening due to crystallite-size and microstrain effects.

RIETAN-94 uses the pseudo-Voigt function of Thompson, Cox & Hastings (1987) modified to some extent. They showed that the Voigt function (David & Matthewman, 1985; Ahtee *et al.*, 1989), *i.e.*, a convolution of the Lorentz function with the Gauss function, can be satisfactorily approximated by a linear combination of the two functions:

$$\phi(\Delta 2\theta) = \eta\phi_{\mathrm{L}}(\Delta 2\theta) + (1-\eta)\phi_{\mathrm{G}}(\Delta 2\theta)$$

$$= \eta\frac{2}{\pi H_k}\left[1 + 4\left(\frac{\Delta 2\theta}{H_k}\right)^2\right]^{-1} + (1-\eta)\frac{2\sqrt{\ln 2}}{\sqrt{\pi}H_k}\exp\left[-4\ln 2\left(\frac{\Delta 2\theta}{H_k}\right)^2\right] \quad (1)$$

with

$$\eta = 1.36603\left(\frac{H_{k\mathrm{L}}}{H_k}\right) - 0.47719\left(\frac{H_{k\mathrm{L}}}{H_k}\right)^2 + 0.11116\left(\frac{H_{k\mathrm{L}}}{H_k}\right)^3 \quad (2)$$

and

$$H_k = \left(H_{k\mathrm{G}}^5 + 2.69269 H_{k\mathrm{G}}^4 H_{k\mathrm{L}} + 2.42843 H_{k\mathrm{G}}^3 H_{k\mathrm{L}}^2 + 4.47163 H_{k\mathrm{G}}^2 H_{k\mathrm{L}}^3 + 0.07842 H_{k\mathrm{G}} H_{k\mathrm{L}}^4 + H_{k\mathrm{L}}^5\right)^{0.2} \quad (3)$$

Here, $\Delta 2\theta = 2\theta_i - 2\theta_k$ (i: step number, k: reflection number, $2\theta_i$: diffraction angle at the ith step, and θ_k: Bragg angle for the kth reflection), ϕ_L is the normalized Lorentz function, ϕ_G is the normalized Gauss function, η is the fraction of the Lorentzian component, H_k is the full-width-at-half-maximum (FWHM) of ϕ_L and ϕ_G, and $H_{k\mathrm{L}}$ and $H_{k\mathrm{G}}$ are respectively the Lorentzian and Gaussian FWHM's for the Voigt function corresponding to the above pseudo-Voigt function.

$H_{k\mathrm{G}}$ is related to the variance of the Gaussian component, σ^2, by

$$H_{k\mathrm{G}} = \sqrt{8\sigma^2 \ln 2} \quad (4)$$

with

$$\sigma^2 = U\tan^2\theta_k + V\tan\theta_k + W + P\sec^2\theta_k \quad (5)$$

Thus, $H_{k\mathrm{G}}$ is a function of the three Gaussian FWHM parameters, U, V, and W (Caglioti, Paoletti & Ricci, 1958), and the Scherrer coefficient, P, for Gaussian broadening. The first term in Eq. (5) contains a component of Gaussian microstrain broadening (Young & Desai, 1989; Von Dreele, 1989). Note that the initial values of U, V, W, and P should give positive σ^2 values over the whole 2θ range.

U, V, and W tend to be highly correlated, with a result that various combinations of quite different values can lead to essentially the same variance, σ^2. These three parameters, therefore, do not converge in a stable manner when refined simultaneously (Prince, 1993). In particular, refining P in addition to U, V, and W almost certainly affords a singular (non-positive definite) coefficient matrix. Of the four profile-shape parameters in Eq. (5), V and W depend not on specimens but only on instruments (Young & Desai, 1989). *Then, these two instrumental parameters may well be fixed at values obtained by the Rietveld refinement of a well-crystallized sample where profile broadening is negligible, i.e., $P = 0$.*

$H_{k\mathrm{L}}$ varies with θ_k as

$$H_{k\mathrm{L}} = (X + X_\mathrm{e}\cos\varphi_k)\sec\theta_k + (Y + Y_\mathrm{e}\cos\varphi_k)\tan\theta_k \quad (6)$$

The first part proportional to $\sec\theta_k$ has the angular dependence associated with Scherrer approximation for crystallite-size broadening, while the second part proportional to $\tan\theta_k$ is related to Lorentzian microstrain broadening. X and Y are isotropic-broadening coefficients (Thompson, Cox & Hastings, 1987; Von Dreele, 1989). On the other hand, X_e and Y_e are anisotropic-broadening coefficients (Larson & Von Dreele, 1990), and φ_k is

the angle between the scattering vector, $\boldsymbol{Q}_k$ (= $h\boldsymbol{a}^* + k\boldsymbol{b}^* + l\boldsymbol{c}^*$), and an anisotropic broadening axis, $h_a\boldsymbol{a}^* + k_a\boldsymbol{b}^* + l_a\boldsymbol{c}^*$.

The parameter P in Eq. (5) provides a component of the Gaussian FWHM which is constant in $\boldsymbol{d}^*$, as is the X component in Eq. (6) (Young, 1993). The modification of the original pseudo-Voigt function of Thompson, Cox & Hastings (1987) consists of the Scherrer coefficient, P, and the anisotropy coefficients, X_e and Y_e (Larson & Von Dreele, 1990). Equations that relate U, P, X, and Y to the crystallite size and microstrain are described in Von Dreele (1989) and Larson & Von Dreele (1990).

Profile asymmetry and peak shift

Profile asymmetry is introduced by employing a multi-term Simpson's rule integration devised by Howard (1982). In this method, n symmetric profile-shape functions with different Simpson's coefficients for weights, g_j, and shifts, f_j, are positioned asymmetrically and superimposed with each other:

$$\phi'(\Delta 2\theta) = \frac{1}{3(n \pm 1)} \sum_{j=1}^{n} g_j \, \phi(\Delta 2\theta') \tag{7}$$

with

$$\Delta 2\theta' = \Delta 2\theta + f_j A_s \cot 2\theta_k + Z + D_s \cos\theta_k + T_s \sin 2\theta_k \tag{8}$$

Here, $\phi'(\Delta 2\theta)$ is the asymmetric pseudo-Voigt function, and $\Delta 2\theta'$ is the 2θ difference modified for (i) profile asymmetry, A_s (Howard, 1982), and (ii) peak shifts for each component profile: zero-point shift, Z, specimen displacement, D_s (Parrish, 1992), and specimen transparency, T_s (Parrish, 1992). The corresponding Simpson's coefficients are:

$n = 3$: $g_1 = g_3 = 1$; $g_2 = 4$

$n = 5$: $g_1 = g_5 = 1$; $g_2 = g_4 = 4$; $g_3 = 2$

$n = 7$: $g_1 = g_7 = 1$; $g_2 = g_4 = g_6 = 4$; $g_3 = g_5 = 2$

$n = 9$: $g_1 = g_9 = 1$; $g_2 = g_4 = g_6 = g_8 = 4$; $g_3 = g_5 = g_7 = 2$

$$f_j = \left(\frac{j-1}{n-1}\right)^2 \tag{9}$$

The number of terms, n (= 3, 5, 7, or 9), in Eq. (7) is either automatically adjusted for each reflection using its FWHM and A_s or fixed at 3 in X-ray powder diffraction using characteristic X rays (NBEAM = 1). The value of A_s must be positive to give reasonable tendencies of peak asymmetry.

The introduction of T_s is indispensable when dealing with X-ray data of compounds with small linear absorption coefficients, μ, which is the case in various organic compounds. Because Z, D_s, T_s, and lattice parameters are highly correlated with each other, refinement of Z, D_s, or T_s in addition to lattice parameters should be carried out very carefully, *e.g.*, by mixing standard reference materials such as Si (NIST, SRM 640b) in samples. These peak-shift parameters sometimes need to be fixed at values refined from intensity data of a standard material by fixing its lattice parameters.

This method of making the profile shape asymmetric gives better fits to asymmetric profiles than the simple one proposed at first by Rietveld (1969), showing less correlation with lattice parameters. Howard's approach further offers physical insight into the origin of the asymmetry because it is based explicitly upon axial divergence. It may, however, fail to fit strongly asymmetric profiles at very low scattering angles. In fact, the Simpson's rule integration can break up into multiple peaks for very strong asymmetry.

Summary

The asymmetric pseudo-Voigt function, $\phi'(\Delta 2\theta)$, composed of Eqs. (1)–(9) contains the nine profile-shape parameters (U, V, W, P, X, X_e, Y, Y_e, and A_s) and the three peak-shift parameters (Z, D_s, and T_s) that can be refined in Rietveld analysis. This function is sound in that it has a physical foundation as well as just fitting the observed diffraction pattern. It can extract microstructural information, *i.e.*, crystallite size and microstrain, from isotropic and/or anisotropic broadening of profiles (Young & Desai, 1989). It can be calculated much faster than the Voigt function (David & Matthewman, 1985; Ahtee *et al.*, 1989). It is also flexible enough to be applicable to conventional X-ray (Post & Bish, 1989), synchrotron X-ray (Cox, 1992), and neutron (Von Dreele, 1989) powder diffraction data.

2.2 Preferred-orientation function

Preferred orientation should be corrected with a function, P_k, which is independent of the diffraction geometry and applicable to both plate and acicular crystallites. Older versions of RIETAN adopted the Sasa-Uda preferred-orientation function (Sasa & Uda, 1976; Toraya & Marumo, 1981):

$$P_k = p_1 + (1 - p_1)\exp(-p_2\phi_k^2) \tag{10}$$

where p_1 and p_2 are refinable parameters, and ϕ_k is the angle between the preferred-orientation direction, $h_p\boldsymbol{a}^* + k_p\boldsymbol{b}^* + l_p\boldsymbol{c}^*$, and the scattering vector, $\boldsymbol{Q}_k$, for reflection k. For $p_1 = 1$, Eq. (10) is equivalent to the Gaussian preferred-orientation function proposed by

Rietveld (1969). When the degree of preferred orientation is small, p_1 should be fixed at 0 because the correlation between p_1 and p_2 becomes too large in such a case.

The March-Dollase function (March, 1932; Dollase, 1986) was added in RIETAN-94:

$$P_k = \frac{1}{m_k} \sum_{j=1}^{m_k} (r^2 \cos^2 \alpha_j + r^{-1} \sin^2 \alpha_j)^{-3/2} \tag{11}$$

where r is an adjustable parameter, and α_j is the angle between the preferred-orientation direction and the jth member of the symmetry-equivalent set of m_k diffraction planes. The March-Dollase function (11) displays the best overall performance for structural studies. It conserves scattering matter, thereby allowing its use in the quantitative analysis of mixtures (Hill & Howard, 1987; Hill, 1993). Note that the mass fractions of constituent phases cannot be determined when the Sasa-Uda function is used to correct for preferred orientation.

The summation over m_k planes and averaging in Eq. (11) are required only when the symmetry is cubic, or the preferred-orientation axis does not lie along a unique axis. In such cases, the scattering vectors of otherwise equivalent planes do not make the same angles. A flag (LSUM) to indicate whether or not the summation should be carried out must be specified in the input file.

The refinable parameter, r, represents the effective sample compression or extension due to preferred orientation. Its value depends on both the diffraction geometry and the crystallite shape:

- Cylindrical sample (*e.g.*, neutron powder diffraction using vanadium cells)

Plate crystallite	$r > 1$
Needle-shaped crystallite	$r < 1$

- Flat-plate sample (*e.g.*, Bragg-Brentano geometry)

Plate crystallite	$r < 1$
Needel-shaped crystallite	$r > 1$

For samples exhibiting no preferred orientation, r is equal to one (*not zero!*).

2.3 Absorption factor

No absorption correction is needed in Bragg-Brentano-type X-ray powder diffraction using flat-plate samples because the absorption factor is constant regardless of 2θ. On the other hand, absorption correction is indispensable in the transmission and Debye-Scherrer geometry, *e.g.*, transmission-speciment methods and neutron powder diffraction

methods using cylindrical containers. Rouse *et al*. (1970) gave an analytical approximation of the absorption factor, $A(\theta_i)$, for cylindrical samples with the radius, r_c:

$$A(\theta_i) = \exp\left[-(a_1 + a_2\sin^2\theta_i)\mu r_c - (a_3 + a_4\sin^2\theta_i)(\mu r_c)^2\right] \tag{12}$$

with a_1 = 1.7133, a_2 = –0.0368, a_3 = –0.0927, and a_4 = –0.3750. Note that there is a printing error for a_4 in the original paper (Rouse *et al*., 1970; Hewat, 1979). The value of the linear absorption coefficient, μ, is calculated from the density of a sample input by the user.

In the transmission specimen method, the absorption factor is

$$A = \left(\frac{t}{\cos\theta}\right)\exp\left(-\frac{s}{\cos\theta}\right) \tag{13}$$

where t is the powder thickness, and s is the sum of the products of the absorption coefficients and thicknesses of the powder and the substrate. We can obtain s easily just by measuring the direct beam intensity with and without a sample. Including this correction allows to obtain more reliable atomic displacement parameters.

2.4 Background function

The background, y_{bi}, at step i is approximated by a finite sum of Legendre polynomials, $F_j(x_i)$ (Abramowitz & Stegun, 1966), orthogonal relative to integration over the interval [–1, 1]:

$$y_{bi} = \sum_{j=0}^{11} b_j F_j(x_i) \tag{13}$$

$F_j(x_i)$'s for $j \geq 2$ are calculated from $F_{j-1}(x_i)$ and $F_{j-2}(x_i)$ using the relation

$$F_j(x_i) = \left(\frac{2j \pm 1}{j}\right)x_i F_{j\pm1}(x_i) \pm \left(\frac{j \pm 1}{j}\right)F_{j\pm2}(x_i) \tag{14}$$

with $F_0(x_i) = 1$ and $F_1(x_i) = x_i$. The coefficients, b_j, are background parameters to be refined in Rietveld analysis, and the variable, x_i, is the diffraction angle, $2\theta_i$, normalized between –1 and 1:

$$x_i = \frac{2\theta_i - \theta_{\max} - \theta_{\min}}{\theta_{\max} - \theta_{\min}} \tag{15}$$

Correlation coefficients between background parameters can be somewhat reduced with this background function. Even "humps" due to amorphous or poorly crystallized compounds may be fit well by increasing the number of refinable background parame-

ters. However, be careful not to vary too many background parameters when dealing with a diffraction pattern whose background has simple dependence on 2θ; refining six background parameters would be adequate in such a case.

3. Input and output

3.1 Files to be input and created by RIETAN-94

Files are named by users or RIETAN-94 according to a definite fashion. If we deal with a compound called 'sample', the following file names are used:

1) sample.ins: input file created by a user.
2) sample.int: X-ray/neutron intensity data (*cf.* 3.4).
3) sample.lst: printer output.
4) sample.xyz: data to be input by ORFFE (Busing, Martin & Levy, 1964) and PRE-TEP (Izumi, 1993).
5) sample.hkl: data to be input by FOURIER (Rotella, 1988).
6) sample.pat: data to plot Rietveld-refinement patterns (observed, calculated, and difference patterns) or simulated ones. There are four different formats for five plotting programs (*cf.* 3.6).
7) sample.ps: graphical data written in the PostScript language (Adobe Systems, 1985) and used to plot Rietveld-refinement/simulated patterns.

Files 1) and 2) are prepared by users, and files 3)–7) by RIETAN-94. Needless to say, ‘sample’ in the above list should be replaced by any other name. ORFFE (calculate interatomic distances, bond angles, *etc.*), PRETEP (a preprocessor for ORTEP-II), ORTEP-II (plot crystal-structure drawings including thermal ellipsoids; Johnson, 1976), and FOURIER (Fourier/D synthesis) are contained in the FAT-RIETAN system (Izumi, 1993). The ORFFE file, sample.xyz, can be converted into two files, sample.mad and sample.atm, using a separate program ffe2am. Sample.mad is an input file for MADEL (calculate electrostatic energies and site potentials), and sample.atm is a file storing atomic parameters for ATOMS (a commercial software of Shape Software for displaying ball and stick models, space-filling models, and coordination polyhedra).

3.2 Input method

A unique input system, F^3IS

Some Rietveld-refinement programs require fixed-column, formatted input data and codewords for parameters contained in the model function (Wiles & Young, 1981; Sakthivel & Young, 1992; Young, Sakthivel, Moss & Paiva-Santos, 1995; Von Dreele, Jorgensen & Windsor, 1982). Such an input manner is too old-fashioned and inconvenient for most users. Interactive menu operations like those in GSAS (Larson & Von Dreele, 1990) and PC Rietveld plus (Fischer *et al.*, 1993) are much more user-friendly but rather troublesome for routine use, making the overlooking of the whole input data nearly impossible because of hierarchic menu structures. In addition, users cannot mod-

ify the content of a menu system at all. RIETAN-94 adopts a novel and creative method of entering data, F^3IS (Flat, Flexible, and Friendly Input System; F cubed S), suitable not only for experts of Rietveld analysis and/or computers but for beginners.

The F^3IS is simple but user-friendly, enabling us to enter data without referring to the manual in nearly all the cases. Input files described in conformation to several rules of the F^3IS are converted into scratch files by a preprocessor called Tink, which was named after a pretty fairy, Tinker Bell, in “Peter and Wendy.” Main advantages of the F^3IS over graphical user interfaces (Fischer *et al.*, 1993) are its portability, straightforwardness, and speedy operation. Tink comprises a few short subroutines, which can be compiled easily with every FORTRAN 77 compiler.

General rules for inputting data consistently

Five rules should be obeyed throughout the input file:

1) One line should be less or equal to 80 columns. If more than 80 columns are input, the extra columns are neglected.
2) When two or more data are input in one line, one or more space(s) should be placed between two data. Never use tabs to separate two data.
3) If the number of data input in one line is variable and has not been determined as yet, the end of input for the line must be indicated by '/' at its tail.
4) When entering a series of lines whose number has not been determined as yet, the end of these lines must be indicated by '}' at the tail of the last line or at the top of the next line.
5) In (i) lines to contain two or more CHARACTER data or (ii) lines where CHARACTER data are mixed with numerical values, CHARACTER data have to be enclosed by a pair of single quotation marks to indicate their ranges.
6) Rule 5) is not applicable if the whole line is read in as a single CHARACTER variable.

Rules 3 and 4 are required so as not to input various numbers of data by users themselves. Lines (positions) where '/' and '}' should be located can be learned by referring to template files.

Comments

With several kinds of template files, we can easily learn which data should be input in each line because many comment lines are included whose first columns are '#'. '#' can be also located after input data; the part from '#' to the end of the line is regarded as a comment. *Two-bytes characters, e.g., kanji, hiragana, katakana, Hankul characters, can be used in comments.* These lines can be freely deleted or modified, and new lines as

well as memoranda may be added by users. Data located separately in two or more lines can be combined together if they are input by list-directed READ statements. It is a disadvantage of menu operations that no such memoranda can be added at all in the screen output.

Lines whose last characters are '{' are regarded as comment lines. Comments may be also placed after '}' without being preceded by '#'. This feature is very convenient when '{' is followed by '}' showing the end of a seaeries of lines. That is, a block surrounded by a pair, '{' (begin) and '}' (end) is made clear as in the C language:

```
Linear constraints for parameters with ID(I) = 2 {
A(GAUSS2,1)=A(GAUSS1,1)
A(GAUSS2,2)=A(GAUSS1,2)
A(GAUSS2,3)=A(GAUSS1,3)
A(LORENTZ2,1)=A(LORENTZ1,1)
A(LORENTZ2,3)=A(LORENTZ1,3)
A(ASYM2,1)=A(ASYM1,1)
A(O2,y)=A(O2,x)
} end of linear constraints
```

Another type of comments is used which follow values of variables and start with colons (:); this will be described below.

Reading both names and values of variables

When the first word in a line is a variable name (INTEGER, REAL, or CHARACTER), '=' plus the value of the variable should follow it. For example, the kind of the angle-dispersive diffraction method is input as

```
NBEAM = 1: Conventional X-ray powder diffraction.
```

'NBEAM' is the variable name, and its value is equal to 1. A colon placed after the value and characters following it (': Conventional X-ray .....') are both optional and regarded as a comment. Tink decodes this line to obtain the value of an integer (*e.g.*, 2), a real (*e.g.*, 3.14159), or a string (*e.g.*, 'Ba'). The name of an integer variable is also stored and referred to in logical expressions in 'If ..... then' described below. The variable name consists of alphabetical capital letters and numbers with the first character being an alphabet. Its maximum length is 10 characters. The first character of an integer variable should be I, J, K, L, M, or N whereas that of a real variable other than I, J, K, L, M, and N.

In the following example, the value of NBEAM is set at 1 by giving '#' at the tops of the other two lines:

```
# NBEAM = 0: Neutron powder diffraction.
  NBEAM = 1: Conventional X-ray powder diffraction.
```

```
# NBEAM = 2: Synchrotron X-ray powder diffraction.
```

If the value of NBEAM is changed, you have to remove one '#' and add one '#', which is somewhat troublesome. We can alternatively set the value of the variable after three comment lines:

```
# NBEAM = 0: Neutron powder diffraction.
# NBEAM = 1: Conventional X-ray powder diffraction.
# NBEAM = 2: Synchrotron X-ray powder diffraction.
  NBEAM = 1
```

Then, we need to only replace the value of the variable. Such a manner of entering a variable value is convenient when it is frequently altered because only one number need to be changed.

We have a convenient way of converting a whole line with the above form into a comment line: putting '!' instead of ':" after the value of a variable. For example, the value of NBEAM is set at 1 in the following way:

```
NBEAM = 0! Neutron powder diffraction.
NBEAM = 1: Conventional X-ray powder diffraction.
NBEAM = 2! Synchrotron X-ray powder diffraction.
```

The first and third lines are regarded as comment lines. The number of '#' in the input file can be reduced greatly according to this manner.

Conditional jump

Pairs of 'If ..... then' and 'end if' serve to make the F^3IS not hierarchic but completely flat. One or more lines may be inserted between 'If ..... then' and 'end if' to form 'if block' in the following manner:

```
If NBEAM = 1 then
   TARG = 'Cu': Radiation ('Cr', 'Fe', 'Co', 'Cu', 'Mo' or 'Ag').
    R12 = 0.5: I(K-alpha2)/I(K-alpha1).
   CTHM = 0.8009: cos(alpha)**2 for the crystal monochromator.
end if
```

The function of the if block is similar to that in FORTRAN 77. Six simple rules have to be followed when using the block if feature.

1) Up to two logical expressions may be placed between 'If' and 'then'.
2) The logical expression have to contain a logical operator, '=', '>', '>=', '<', '<=', or '<>' (≠).

3) Two logical expressions are related to each other with a relational operator, 'and' or 'or', *e.g.*, 'If NMODE=0 and NPAT=1 then'.
4) When the condition described between 'If' and 'then' is false (not satisfied), the if block (in the above cases, the three lines) between the 'If ..... then' and 'end if' lines is skipped automatically.
5) If the condition is true (satisfied), the if block is read in by the program.
6) Unlike FORTRAN 77, nesting of 'If ..... then' and 'end if' pairs is not allowed to keep simplicity.

The if block feature is most convenient because we can leave lines that would otherwise be attached with '#' or deleted. Someone may be afraid that this feature is not easy to understand for most users . However, I believe that those who cannot comprehend this concept will never be able to carry out Rietveld refinements, too.

How to input labels, parameters, and refinement identifiers

Labels, parameters contained in the model function (background, profile, structure parameters, *etc*., refer to Supplement), and their refinement identifiers (unnecessary in the simulation of powder diffraction patterns) are input according to the following rules:

1) A label is located at the top of a line without any preceding space, followed by a group of parameters and corresponding refinement identifiers.
2) A label consists of alphabetical letters (a–z and A–Z) and numbers (0–9), but *its first character should be a capital letter*. Its maximum length is 25 characters (less than 8 characters are desirable for the sake of printing).
3) The label need not be enclosed by a pair of single quotation marks.
4) A label can be arbitrarily assigned to a group of parameters except that one label must be always assigned to one crystallographic site by grouping all the structure parameters for the site. The parameters are hereafter managed under the name of the label.
5) A label for an atomic site is attached with '/' plus the name of a chemical species without inserting any space. The term 'chemical species' denotes a real species included in the database file, asfdc, or a virtual species derived from two or more real species for a mixed-atom site. In neutron diffraction, use only neutral species (neither cations nor anions). For magnetic atoms, '*' must be attached to chemical species names.
6) Two or more lines may be used for parameters grouped under a label.
7) If a dummy '+' sign is attached to an isotropic atomic displacement parameter, B, in such a way as '+1.2', the program automatically converts it into six anisotropic atomic displacement parameters, β_{ij}. On the other hand, the refinement identifiers

and linear constraints imposed on the anisotropic atomic displacement parameters (Peterse & Palm, 1966) must be input by the user. The number of the refinement identifiers is 10 in this case.

8) If a *B* value other than zero is input, and five zero values follow it as dummy parameters, six β_{ij} values are calculated from *B* in a similar way as in 7). This feature is indispensable when NUPDT = 3.
9) Refinement identifiers for fixed, varied, and constrained parameters are 0, 1, and 2, respectively. 'Constrained' means that a parameter whose refinement identifier is equal to 2 is evaluated from other refinable parameter(s) with a linear equality constraint.
10) A group of refinement identifiers is input after structure parameters without inserting any space among them.
11) Even if comments are attached just after refinement identifiers, they are deleted when parameters in the input file are updated to refined ones. On the other hand, comment lines inserted between these parameter lines are reserved on updating of parameters.

For instance

```
Fe/Fe3+ 1.0 0.3459 0.3459 0.5 0.6 01201
```

is a line input for a metal site. 'Fe' is the label for this site, 'Fe3+' is the name of a chemical species, '1.0' is the occupation factor (*g*), '0.3459 0.3459 0.5' are fractional coordinates (*x*, *y*, and *z*), '0.6' is the isotropic atomic displacement parameter (*B*), and '01201' is the refinement identifiers for the five structure parameters. The *y* coordinate is constrained to be equal to the *x* coordinate.

Labels can be conveniently used when referring to (i) *parameters* in linear equality constraints and (ii) *serial numbers for parameters* varied in early refinement cycles. Each parameter number may be represented as 'L,I' or 'L,S', where 'L' is a label to which the parameter belongs, 'I' is the parameter number within the group of parameters under label 'L', and 'S' is the symbol of a structure parameter. Each parameter is then represented as 'A(L,I)' or 'A(L,S)' with the array name 'A' and a parameter number in the parenthesis. 'A(L,I)' may be represented as 'A(L)' if I is 1.

For structure parameters, we can use symbols, 'g' (occupation factor, *g*), 'x', 'y', 'z' (fractional coordinates, *x*, *y*, and *z*), 'B' (isotropic atomic displacement parameter, *B*), 'beta11', 'beta22', 'beta33', 'beta12', 'beta13', and 'beta23' (anisotropic atomic displacement parameters, β_{ij}), instead of their numbers. We may replace 'beta' with 'B'; for example, 'B12' is equivalent to 'beta12'. In the case of the line 'Mg/Mg2+ 1.0 0.345 0.0 0.5 0.8 01001', 'A(Mg,x)' and 'A(Mg,B)' are respectively the *x* coordinate and isotropic atomic

displacement parameter for the Mg site. 'A(Mg,x)' and 'A(Mg,B)' may be alternatively expressed as 'A(Mg,2)' and 'A(Mg,5)', respectively.

Linear equality constraints

Linear equality constrains can be imposed on profile and structure parameters. They are described in a similar manner as assignment statements in FORTRAN except that each parameter is represented as A(L,I) or A(L,S) as described above and that '/' should not be included in the right side. 'A(Gauss2,1)=A(Gauss1,1)', 'A(Asym2,1)=A(Asym1,1)', A(Fe,y)=2.0∗A(Fe,x)', 'A(Co1,g)=1.0 –0.5∗(Cu2,g)', and 'A(O2,B12)=0.5∗A(O2,B22)' exemplify a coordinate, an occupancy, and an anisotropic atomic displacement parameter, respectively. The parameter in the left side has always a refinement identifier of 2 and constrained in such a way that the above equation is strictly satisfied. The right side includes at least one refinable parameter but no fixed parameters. In the above constraint on g, only g(Cu2) is refined, and g(Co1) is calculated from g(Cu2) with the above constraint. Of course, the above linear relation is taken into account in the calculation of the partial derivative of the model function with respect to g(Cu2). As in the above example, linear constraints on anisotropic atomic displacement parameters (Peterse & Palm, 1966) should be input by the user.

We can describe two or more linear constrains in one line, marking off by semicolons (;) as follows:

```
A(Fe,y)=2.0*A(Fe,x); A(N2,g)=1.0–0.5*A(O2,g); A(O2,B)=A(O1,B)
```

The number of linear constraints is equal to that of parameters whose refinement identifier is equal to 2.

Linear constrains on profile parameters are used in such a way that profile parameters of impurity phases are set equal to those of the main phase. Imposing such constrains considerably reduces the total number of profile parameters, in particular, in samples containing three or more phases.

Parameters refined in each cycle

When NAUTO is 1 (incremental refinement), parameters refined in initial cycles are customized by users. Parameter numbers are input by using L,I and/or L,S and separating two numbers with one or more space(s). '/' at the tail of a line indicates the end of refinable parameters in a refinement cycle. If a line does not end with '/', parameters in the next lines follow those in the present line. '}' is placed at the end of a series of lines. For example, if we input

```
BKGD,1 BKGD,2 BKGD,3 BKGD,4 BKGD,5 BKGD,6 BKGD,7 BKGD,8 SCALE,1/
CELL,1 CELL,2 CELL,3/
PRFL,1 PRFL,2 PRFL,3 PRFL,5 PRFL,7 PRFL,9/
```

```
Ti,x Ti,B O1,y O1,z O1,B O2,g O2,B/
}
```

eight background parameters and a scale factor are refined in the first cycle, three lattice parameters, *a*, *b*, and *c*, in the second cycle, six profile-shape parameters in the third cycle, and seven structure parameters in the fourth cycle. In subsequent cycles, all the parameters whose ID(I)'s are 1 are refined simultaneously. If NAUTO is 2, RIETAN-94 automatically specifies appropriate combinations of refinable parameters for several cycles.

3.3 Notes on some input data

Magnetic form factors

In RIETAN-94, magnetic form factors are calculated from seven coefficients (*A*, *a*, *B*, *b*, *C*, *c*, and *D*) in an analytical approximations to the $\langle j_0(s) \rangle$ magnetic form factors (l = 0) as a function of s (= $\sin\theta/\lambda$):

$$\langle j_0(s) \rangle = A\exp\left(-as^2\right) + B\exp\left(-bs^2\right) + C\exp\left(-cs^2\right) + D \qquad (16)$$

The coefficients in Eq. (16) are listed in International Tables, Vol. C (1992) for the 3d and 4d transition series, the 4f electrons of lanthanoid ions, and the 5f electrons of some actinoid ions. In an input file (∗.ins), *l (= 0), A, a*, *B*, *b*, *C*, *c*, and *D* are entered for a chemical species whose name is attached with '∗' in the following way:

```
# The following line is input for Fe2+ (l = 0).
0 0.0263 34.960 0.3668 15.943 0.6188 5.594 -0.0119
```

Real and virtual species

Real species denote neutral atoms, cations, and anions whose various physical quantities are stored in the database file asfdc. To learn their names, refer to asfdc. On the other hand, virtual species are those composed of two or more real species. The use of such an imaginary species is very convenient when dealing with compounds where more than two kinds of atoms occupy the same site, decreasing the number of structure parameters and linear constraints imposed on them. Names and amount-of-substance fractions of the constituents real species must be given by the user. For example, for a virtual species named 'LaCa' consisting of 90% of La and 10% of Ca, we input

```
'LaCa' 'La' 0.9 'Ca' 0.1/
```

Names of both real and virtual species may be input after label + '/' in structure-parameter lines.

Two methods of estimating standard deviations

In Rietveld analysis, the estimated standard deviation (e.s.d.) is usually calculated in the same way as in single-crystal structure refinement (Young, 1993). If serial correlation is present, no e.s.d. is a valid measurement of uncertainty. Only if the model is completely correct, which implies that any systematic errors in the data must be appropriately described, least-squares error estimates are reliable indicators of accuracy (Post & Bish, 1989). Scott (1983) proposed converting Rietveld e.s.d.'s to equivalent integrated intensity deviations on the grounds that these better reflect the effects of model errors. He conceded, however, that the statistical basis of this conversion is dubious and that it should apply to only structure parameters. RIETAN-94 provides an option where we can select the method of calculating .e.s.d.'s: conventional and Scott's methods.

Nonlinear constraints for interatomic distances and bond angles

In RIETAN-94, nonlinear constrains are imposed on (i) interatomic distances, l_{12}, for Atoms 1 and 2 and (ii) bond angles, ϕ_{123}, for Atoms 1, 2, and 3 with Atom 2 as the apex by adding penalty terms

$$t^{(K)}\sum_j \frac{1}{l_{12j}(\mathrm{cal})}\left[\min\left(0,\ \Delta l_{12j} - \left|l_{12j}(\mathrm{cal}) - l_{12j}(\mathrm{exp})\right|\right)\right]^2 +$$

$$t^{(K)}\sum_k \frac{1}{\phi_{123k}(\mathrm{cal})}\left[\min\left(0,\ \Delta\phi_{123k} - \left|\phi_{123k}(\mathrm{cal}) - \phi_{123k}(\mathrm{exp})\right|\right)\right]^2 \qquad (17)$$

to the weighted sum-of-squares, $S(\boldsymbol{x})$. Then, the sum of $S(\boldsymbol{x})$ and the penalty terms is minimized by a method of nonlinear least squares in Rietveld analysis. In Eq. (17), $t^{(K)}$ is the penalty parameter in the Kth stage of unconstrained minimization (Izumi, 1993), l_{12j} (cal) is l_{12} calculated from the current structure and lattice parameters for the jth atom pair, l_{12j} (exp) is l_{12} expected for the same pair, Δl_{12j} is the allowance for l_{12j}, ϕ_{123k}(cal) is ϕ_{123} calculated from the current structure and lattice parameters for the kth atom trio, ϕ_{123k}(exp) is ϕ_{123} expected for the same trio, and $\Delta\phi_{123k}$ is the allowance for ϕ_{123k}. Interatomic distances lying in the range l_{12j}(exp) ± Δl_{12j} are never penalized. Penalties are likewise not imposed on bond angles within the range ϕ_{123k}(exp) ± $\Delta\phi_{123k}$. In the present version, penalties for l_{12j} and ϕ_{123k} are multiplied by weights of $1/l_{12j}$(cal) and $1/\phi_{123k}$(cal), respectively, to deal with l_{12} and ϕ_{123} having a wide range of values roughly evenly.

In older versions of RIETAN, nonlinear constrains must be described in FUNCTION CON; then, recompiling and linking are required to make it possible to introduce

nonlinear constraints into Rietveld refinements. In RIETAN-94, all that users have to do is to enter series numbers for l_{12} and ϕ_{123} in an output file, *.ffe, of ORFFE (Busing, Martin & Levy, 1964), l_{12j}(exp) or ϕ_{123k}(exp), and Δl_{12j} or $\Delta\phi_{123k}$ in the input file (*.ins). Example lines are shown below where two C–O distances and two O–C–O bond angles within a triangular CO_3^{2-} ion in $Ba_4CaCu_2O_6CO_3$ are confined within the ranges 1.29±0.025 Å and 120±3°, respectively (Kikuchi *et al.*, 1995).

```
If NC = 1 then
   # Ser.No.  Expected.val.  Allowance
      59         1.29         0.025
      67         1.29         0.025
     134       120.0          3.0
     135       120.0          3.0
}
end if
```

An output file of ORFFE (*.ffe) need to be created prior to constrained refinement because several kinds of data (series numbers, symmetry operations, l_{12j}(cal), ϕ_{123k}(cal), *etc*.) included in it are read in by RIETAN-94. Then, at least one unconstrained Rietveld refinement should be carried out prior to constrained one to obtain *.ffe. It should be noted that *.ffe can be created only when it does not exist in the same folder as *.xyz. *If you want to update *.ffe, you must remove it prior to the execution of ORFFE.* In *.xyz files, '201' functions to calculate interatomic distances must be always included with optional '2' functions to calculate bond angles.

RIETAN reads in symmetry operations and translations from *.ffe and apply them to fractional coordinates of atoms related to intertatomic distances and/or bond angles with the serial numbers specified by users. This technique of introducing nonliear constraints with a reference file output by ORFFE is referred to as 'Eleonora,' which is named after a mysterius short novel by E. A. Poe.

ORFFE functions

A number of functions is available in ORFFE (Busing, Martin & Levy, 1964). However, the following four functions would be sufficient in nearly all structure refinements:

Function 201

All distances (less than l_{max}) between A_{max} atoms in the asymmetric unit and atoms in all asymmetric units, i.e., all combinations of C and S. For details in A, C, and S, refer to an output of ORFFE.

Columns

1–3	201
4–6	A_{max}, the number of atoms in the asymmetric unit.
16–18	The integer $10l_{max}$. If this is left blank, then l_{max} = 4.0 Å.

The format of this line is (2I3,9X,I3).

Function 2

Bond angle defined by three atoms.

Columns

1–3	2
4–9	Atom designation 1.
10–15	Atom designation 2 (vertex).
16–21	Atom designation 3.

The format of this line is (7I3). Each atom designation consists of A and 100C+S (3 columns for each).

Function 1

One interatomic distance.

Columns

1–3	1
4–9	Atom designation 1.
10–15	Atom designation 2.

The format of this line is (5I3). Each atom designation consists of A and 100C+S (3 columns for each).

Function 101

All distance (less than l_{max}) between atoms in two asymmetric units.

Columns

1–3	101
4–6	A_{max}, the number of atoms in the asymmetric unit.
7–9	$100C_1 + S_1$, designation of the first asymmetric unit.
10–12	—
13–15	$100C_2 + S_2$, designation of the second asymmetric unit (this may be the same as the first.)

16–18 The integer $10l_{max}$. If this is left blank, then l_{max} = 4.0 Å.

The format of this line is (3I3,3X,2I3).

These four functions, if any, must be ordered as above.

3.4 Update in database files

Atomic weights, bound scattering lengths and cross sections for neutrons, and anomalous dispersion corrections in the database file, asfdc (Izumi, 1993), were updated to the latest values compiled in "International Tables," Vol. C (1992a). *Capital and small letters are now differentiated from each other in both input and database files: asfdc, spgri, and spgra* (Izumi, 1993). For example, not 'FE' but 'Fe' must be input as a name of the element, iron. Output lists are likewise represented using capital and small letters, which improves their readability considerably.

3.5 New formats of intensity data

The so-called RIETAN format of intensity data is as follows: (i) comment lines with '*' at the first column (may be skipped), (ii) number of data points, starting 2θ, and step width, and (iii) intensities in subsequent lines. One or more spaces are placed between two data.

RIETAN-94 can deal with text files of intensity data created by the latest measurement/control programs for Rigaku and MAC Science X-ray powder diffractometers (do not delete any lines when dealing with files with these two formats).

RIETAN-94 can also read in 'general-format' files storing intensity data with variable step widths of 2θ. In a file conforming to this format, the first line should be 'GENERAL', the second line the number of data points, and the remaining lines 2θ and intensity pairs.

Intensity-data files with the Igor text file format can be read in by RIETAN-94. They must have the following format:

```
IGOR
WAVES twoth, yobs
BEGIN
```
Pairs of 2θ and observed intensities are input repeatedly here.
```
END
```

Neutron powder diffraction data measured on the following diffractometers can be also input directly: (i) HRPD diffractometer installed at the JRR-3M reactor at the Japan Atomic Energy Research Institute (Morii *et al.*, 1992), (ii) D2B diffractometer at the Institut Laue Langevin, and (iii) HB-4 Spectrometer at the HFIR reactor at Oak Ridge

National Laboratory. In (ii) and (iii), the first line of each intensity data file must have 'D2B' and 'HB-4' as the first for characters, respectively.

Some instruments, such as D2B and HB-4, produce data where different numbers of detectors, n_i, contribute to the intensity data, y_i, recorded at various steps. In such cases, a weight table is generated in which the weight at each step, i, is calculated as

$$w_i = n_i / y_i \tag{18}$$

3.6 New output data

The Durbin-Watson d statistic (Hill & Madsen, 1986; Hill & Flack, 1987) is printed out in each refinement cycle. It is very useful for assessing the reliability of estimated standard deviations in Rietveld analysis, by providing us with quantitative information about serial correlation in the residuals. It is also a sensitive measure of the progress of a refinement, and is still discriminating even when other reliability indices fail. Estimated standard deviations are given for unic-cell volumes (Giacovazzo, 1992). For samples containing two or more phases, the mass fractions of constituent phases calculated from their scale factors (Hill & Howard, 1987; Hill, 1993) are printed out directly. However, it should be noted that, in its current form, the mass fraction is calculated without any correction for microabsorption (Taylor & Matulus, 1991), which is particularly serious in X-ray powder diffraction. Please correct for microabsorption if accurate mass fractions are required.

If NPRINT is equal to 2, two impressive phrases (Young, 1993; Prince, 1981) are finally printed out which must be very useful as guidelines for Rietveld analysis:

"*We should note that stopping and finishing are not the same thing. The refined model must make physical and chemical sense, or it is not finished ... and even then it might be wrong!*"

R. A. Young, 1993

"*If the fit of the assumed model is not adequate, the precision and accuracy of the parameters cannot be validly assessed by statistical methods.*"

E. Prince, 1981

We should analyze intensity data, always keeping these two epigrams in mind.

3.7 Files to plot Rietveld-refinement and simulated patterns

RIETAN-94 has a feature to create text files storing data for drawing observed, calculated, and difference patterns (so-called Rietveld-refinement patterns) in five different formats: (i) PostScript (Adobe Systems, 1985), (ii) Macplot and RietPlot, (iii) PLOT

(Sakthivel & Young, 1992), (iv) SigmaPlot (Jandel Scientific), and (v) Igor text file (WaveMetrics, 1994). As described in 3.1, file (i) has a name ending in '.ps' whereas files (ii)–(v) have names ending in '.pat'. No simulated patterns can be recorded in file (iii). Files (ii)–(v) are created in subroutines OUT20 (Rietveld analysis) and SIMDAT (simulation); files with another format can be easily obtained if some simple codes are added in this subroutine by referring to comment lines.

PostScript

Graphical data recorded in a PostScript file can be output to screens using either previewers which are included in most UNIX workstations or GhostScript, a famous freeware provided by the GNU project. If a printer equipped with a PostScript interpreter is available, figures can be directly printed out, *e.g.*, by entering 'type sample.ps > prn' on MS-DOS machines and by using (i) the 'Download PostScript File' command under the Utilities menu in 'LaserWriter Utility' or (ii) Drop·PS on the Macintosh. In the current version, tick marks are plotted for up to two phases.

Macplot/RietPlot

There are two interactive programs to plot Rietveld-refinement and simulated patterns using files with the original format. One is a FORTRAN program named Macplot for the Macintosh. The other is a Windows program, RietPlot, coded in Visual Basic by Ohashi. Both of them support outputs to any types of printers.

PLOT

PLOT is a shareware attached to the Rietveld-refinement program DBWS-9006PC developed by Sakthivel & Young (1992). This program for IBM-PCs supports high-resolution output to HP LaserJet and PostScript page printers. It is a matter for regret that PLOT can deal with only refinement results where the step width is strictly constant over the whole 2θ range. For example, it fails in plotting results of Rietveld refinements using intensity data measued on the HRPD of JAERI. Its another disadvantage is that it cannot be run in the DOS prompt mode under Microsoft Windows.

Files with the PLOT format can be also read in by DMPLOT (Ver. 3.47 as of this writing) developed by Marciniak (1995). Please note that DMPLOT also requires a constant step width over the whole 2θ range.

SigmaPlot

SigmaPlot (Jandel Scientific, 1994) is a popular, commercial program which can be run on the MS-DOS, Microsoft Windows, and Macintosh for graphing and data analysis. It enables us to obtain publication-quality Rietveld-refinement/simulation patterns

with any user-desired size, full scale, and 2θ region using any kind of printers. With SigmaPlot, drawing the patterns is rather tedious and time-consuming. However, the feature of a page template considerably reduces times to draw similar patterns.

SigmaPlot-format files created by RIETAN-94 for Rietveld-refinement patterns have the following data which occupy columns of a data worksheet:

1) Diffraction angles, $2\theta_i$.
2) Observed intensities, y_i.
3) Calculated intensities, $f_i(\boldsymbol{x})$.
4) Differences between the observed and calculated intensities, $y_i - f_i(\boldsymbol{x})$. Decreased appropriately for clarity.
5) Half the length of tick marks indicating peak positions.
6) X coordinates of tick marks, *i.e.*, peak positions, $2\theta_k$.
7) Y coordinate of tick marks.

Data 6 and 7 are repeated by numbers of phases. Data 4), 5), and 7) have been determined temporarily by RIETAN-94. In most cases, their values need to be changed using the Math menu, Transform command. Data 5)–7) are repeated by the number of phases but up to three phases. In the case of simulation patterns, data 1), 3), 5), 6), and 7) are output for only one phase.

To read in a SigmaPlot-format file, choose the File menu, Import Data... comand, select to import Plain Text files in the List Files of Type box, and then select the file. Fixed culumns should be specified as Field format.

For convenience, tick marks need to be plotted as error bars without any symbols or caps. Observed intensities can be best shown as crosses using the following setting of symbols: Symbol Type: Symbol (none); Fill Color: Color (none); Appearance: Crosshair (enter an appropriate size); Edge Color: Color (enter an appropriate color); Edge Thickness: (select an appropriate thickness). Extra negative tick labels (negative intensities may appear in the lower part to plot peak positions and a difference pattern) and ticks in the ordinate can be skipped using the feature of customizing tick labels (*cf.* 9-27 in the user's manual).

Igor text file

Igor Pro for the Macintosh is a graph-plotting program similar to SigmaPlot but more powerful than SigmaPlot when drawing two-dimensional plots. If a DOS or Windows version of Igor Pro were available, no SigmaPlot format would be supported in RIETAN-94. Igor text files output by RIETAN-94 include not only refinement results but a series of commands to produce the patterns automatically. Igor Pro is, therefore, more rapid and convenient for plotting Rietveld-refinement/simulation patterns than SigmaPlot. Refer to Wave-Metrics (1994) to learn details in Igor text files.

To plot Rietveld-refinement/simulation patterns, all what you must do is (i) to select 'Load Igor Text' in the 'Load Waves' submenu under the 'Data' menu and then specify a '*.pat' file to be read in, (ii) to double-click this file from the Finder, or (iii) to drag this file and drop onto an icon of Igor Pro. Smart operations like (ii) and (iii) are possible because Igor text files created by RIETAN-94 hold information on the Type and Creator characteristics of Macintosh-format files. Tick marks are plotted for up to three phases. Extra negative tick labels and corresponding ticks for the ordinate are erased using the feature of 'Enable ticks between' in the 'Tick Tweaks' sub-dialog under the 'Modifying Axes' item from the 'Graph' menu. The following wave names are given for various kinds of data:

twoth	$2\theta_i$
yobs	y_i
ycal	$f_i(\boldsymbol{x})$
delta	$y_i - f_i(\boldsymbol{x})$
xphase_*n*	peak positions (2θ) for phase *n*
yphase_*n*	*y* coordinates of tick marks for phase *n*
d_n	lattice-plane spacings for phase *n*
h_*n*	*h* indices for phase *n*
k_*n*	*k* indices for phase *n*
l_*n*	*l* indices for phase *n*

You can learn lattice-plane spacings and indices of reflections by specifying their wave names creating a table ('New Table' under the 'Windows' menu); select 'Edit data columns only' in this case.

A pattern of only one phase can be plotted in the case of simulation, as in SigmaPlot. Vertical positions of difference patterns and tick marks are adjusted by inputting commands in the command window in such a way as 'delta=delta+ 300' and 'yphase1= yphase1–120'.

With Igor Pro, we can also scale part of a graph manually (refer to the manual, p. 228). Click the mouse and drag it diagonally to frame the region of interest. When you click the region of interest, Igor Pro presents a popup menu from which you can choose any of the scaling operation. These operations can be undone and redone by typing command-Z.

X and *y* coordinates of a point can be read in the following way (*cf*. User's manual, p. 267):

1) Select the 'Show Info' item from the "Graph' menu. The alternative to this operation is to press Command-I.
2) Click a cursor at home (two cursors, circle with + and square with ×, are available) and drag it to the desired point on the wave whose values you want to know.

3) *X* and *y* values appear in the readout area.

4) The cursor can be moved by (a) dragging the slide control right or left, (b) clicking on one side or the other of the slide control, (c) using arrow keys on your keyboard (particularly convenient when examining yphase*n*), or (d) dragging the cursor directly to the desired point. Note that the cursor can be moved to another wave.

5) If the cursor is no longer necessary, you can remove it from the graph by dragging it away from the plotting area.

6) Select the 'Hide Info' item from the "Graph' menu. The alternative to this operation is to press Command-I.

4. Other alternations

4.1 Maximum array dimensions

Integer constants to set maximum dimensions of many arrays are defined in PARAMETER statements. They were increased to considerably larger values in RIETAN-94 than in the previous version, RIETAN-9210: NP (data points) = 17000, NPH (constituent phases) = 8, NB (reflections) = 3100, NAP (atoms in the asymmetric unit) = 96, NT (parameters contained in the model function) = 515, NR (refinable parameters) = 130, NSF (refinable structure parameters) = 100, and NCS (linear equality constraints) = 70. These values are so large that we seldom need to increase them to adapt RIETAN-94 for specific refinements.

4.2 Bug fixing

Several bugs in the previous version (RIETAN-9210) as well as β versions of RIETAN-94 have been fixed in RIETAN-94, including (i) failure to skip if blocks in some special cases, (ii) missing part of possible reflections on generation of reflections (this has not perfectly been fixed as yet), (iii) failure to select essential symmetry operations by checking the overlap of atoms, (iv) neglecting profile asymmetry on determining the profile cut-off for each reflection, (v) failure in calculating the preferred-orientation functions after the first cycle in the Marquardt and Gauss-Newton Methods, (vi) failure in updating parameters properly when NUPDT = 2, and (vii) occasional addition of extra lines at tails of input files when NUPDT = 1 (in the Macintohs version only). The author strongly hopes that bugs in RIETAN-94 are reported to him immediately after they have been found.

5. Desktop or laptop Rietveld analysis

RIETAN-94 was originally developed on a SPARCstation 2 workstation. However, a new generation of personal computers (PC) with Pentium and PowerPC processors has already made such workstations inessential (Tobochnik & Tobochnik, 1993), at least, for conventional crystallographic computation. These PCs differ from the workstations in the point that they are 5 to 10 times cheaper while offering equal or higher performance. Their high cost performance results from the mass-market nature of PCs, by contrast with the low volume sales of the workstations. Another serious disadvantage of workstations is that there are more than 12 different versions of UNIX operating systems which are not compatible with each other. Furthermore, the cost of mainframes and minicomputers would be much higher than workstations.

Why don't you use PCs for Rietveld analysis instead of expensive mainframes, minicomputers, and workstations? In particular, compatibles of the IBM-PC/AT are surprisingly cheap. Scientists who lack in research expenses can save much money by running the free software, RIETAN-94, on these popular computers.

RIETAN-94 'for the rest of us' is currently available for (i) MS-DOS (PC-DOS) machines including IBM-PCs, NEC PC-9801s (domestic PCs in Japan), and their compatibles and (ii) Apple Macintosh computers. Those who hate editors suitable for programmers (*e.g.*, emacs) must enjoy easy editing with familiar editors for PCs coupled with the F^3IS. Because editors for the Mac OS and Windows generally conform to their definite rules, we need not learn their basic operations. Another advantage of the PC versions is that results of Rietveld refinements can be plotted with the five different programs described in 3.6.

The DOS version was developed using NDP FORTRAN-Pentium (Ver. 4.5.1) and adapted to run it in protected mode in combination with the NDP/Ergo SDK DOS Extender, which practically removes the limitation on the maximum main memory of 640 Kbytes in MS-DOS. Computers equipped with 80386+80387, 80486, or Pentium processors are required to run it. This stand-alone program can run on both DPMI and VCPI in the DOS prompt mode under Microsoft Windows. It is quite puzzling that the executable binary file optimized for the Pentium is somewhat *slower* than that for the 80486 processor, which is the reason for distributing only the executable file optimized for 80486 at present.

The source program of RIETAN-94 has been compiled with Language Systems FORTRAN for Power Macintosh (Ver. 1.2) to generate a Power Application, which runs in native mode on the Power Macintosh. A dramatic increase in execution speed has been attained with the Power Macintosh native version thanks to the innovative RISC architecture of the PowerPC, in particular, its enhanced ability in floating-point calcula-

tions. Power Macintoshes equipped with 604 processors can run PowerPC native codes about 1.5 times as fast as those containing 601 processors with a comparable clock rate.

RIETAN-94 can be launched by (a) double-clicking the icon of RIETAN-94 and specify an input file, (b) double-clicking the icon of RIETAN-94 or an input file (*.ins), or (c) drag the icon of the input file onto the that of RIETAN-94. When using feature (a), the creator and type of the input file must be respectively changed into 'RIET' and 'TEXT' with a utility such as File Buddy. Output windows are opened immediately after launching RIETAN-94, displaying output lists on the screen. After the execution, the windows can be scrolled to see the output, which can be edited and/or printed out using pull-down menus. This is an excellent feature which is not supported in the DOS version.The author will use not the SPARC-station 2 version but the PowerPC version for routine Rietveld analysis in combination with the amazing program, Igor Pro (*cf.* 3.6).

6. To users of RIETAN-94

Everyone in every country may use RIETAN-94 without any restriction on the conditions that the author is not responsible for results obtained with it and that the following reference is quoted in papers which report crystal data obtained with it:

F. Izumi, "The Rietveld Method," ed. by R. A. Young, Oxford University Press, Oxford (1993), Chap. 13.

RIETAN-94 is distributed free of charge to spread and develop powder diffraction techniques. It may be copied and given away to anyone without any permission of the author. The author has written this manual not in Japanese but in English, hoping RIETAN-94 to be widely used in the world. He would be much honored if RIETAN-94 could serve excellent scientific work. Sending him reprints of papers including crystal data obtained with RIETAN-94 would greatly encourage him in developing a forthcoming version of RIETAN.

Acknowledgments

The author wishes to thank Miss Y. G. for providing him with motivation and activity to update RIETAN during his short stay in Peking in August, 1993. R. B. Von Dreele kindly informed the author of details in the profile-shape function adopted in GSAS. Thanks are also due to N. Ohashi for developing the pattern-drawing program, RietPlot, for Microsoft Windows, T. Kamiya for his help in developing the DOS version of RIETAN-94, and Y.-I. Kim for updating the database files.

References

Abramowitz, M. & Stegun, I. A. (1966). "Handbook of Mathematical Functions," Applied Mathematics Series 55, National Bureau of Standards, pp. 771–802.

Adobe Systems (1985). "PostScript Language, Reference Manual," Addison-Wesley, Reading, Massachusetts.

Ahtee, M., Nurmela, M., Suortti, P. & Jävinen, M. (1989). *J. Appl. Crystallogr.*, **22**, 261.

Albinati, A. & Willis, B. T. M. (1982). *J. Appl. Crystallogr.*, **15**, 361.

Altomare, A., Cascarano, G., Giacovazzo, C., Guagliardi, A., Burla, M.C., Polidori, G. & Camalli, M. (1994). *J. Appl. Crystallogr.*, **27**, 435.

Busing, W. R., Martin, K. O. & Levy, H. A. (1964). "A FORTRAN Crystallographic Function and Error Program," Report ORNL-TM-306, Oak Ridge National Laboratory, Oak Ridge, Tennessee.

Caglioti, G., Paoletti, A. & Ricci, F. P. (1958). *Nucl. Instrum. Methods*, **3**, 223.

Cheetham, A. K. & Taylor, J. C. (1977). *J. Solid State Chem.*, **21**, 253.

Cox, D. E. (1992). "Synchrotron Radiation Crystallography," ed. by P. Coppens, Academic Press, London, pp. 196–199.

David, W. I. F. (1988). "Neutron Scattering at a Pulsed Source," ed. by R. J. Newport, B. D. Rainford & R. Cywinski, Adam Hilger, Bristol, Chap. 12.

David, W. I. F. & Matthewman, J. C. (1985). *J. Appl. Crystallogr.*, **18**, 461.

Dollase, W. A. (1986). *J. Appl. Crystallogr.*, **19**, 267.

Fischer, R. X., Lengauer, C., Tillmanns, E., Ensink, R. J., Reiss, C. A. & Fantner, E. J. (1993). *Mater. Sci. Forum*, **133–136**, 287.

Giacovazzo, C. (1992). "Fundamentals of Crystallography," ed. by C. Giacovazzo, Oxofrd University Press, Oxford, p. 122.

Hall, Jr., M. M., Veeraraghavan, V. G., Rubin, H. & Winchell, P. G. (1977). *J. Appl. Crystallogr.*, **10**, 66.

Hewat, A. W. (1979). *Acta Crystallogr., Sect. A*, **35**, 248.

Hill, R. J. (1993). "The Rietveld Method," ed. by R. A. Young, Oxford University Press, Oxford, Chap. 5.

Hill, R. J. & Flack, H. D. (1987). *J. Appl. Crystallogr.*, **20**, 356.

Hill, R. J. & Howard, C. J. (1987). *J. Appl. Crystallogr.*, **20**, 467.

Hill, R. J. & Madsen, I. C. (1986). *J. Appl. Crystallogr.*, **19**, 10.

Howard, C. J. (1982). *J. Appl. Crystallogr.*, **15**, 615.

"International Tables for Crystallography," Vol. C (1992). Kluwer, Dordrecht, pp. 391–399.

"International Tables for Crystallography," Vol. C (1992a). Kluwer, Dordrecht, pp. 219–222 and pp. 384–391.

Izumi, F. (1989). *Rigaku J.*, **6**, No. 1, 10.

Izumi, F. (1993). "The Rietveld Method," ed. by R. A. Young, Oxford University Press, Oxford, Chap. 13.

Izumi, F. (1993a). *Trans. Am. Crystallogr. Assoc.*, **29**, 11.

Izumi, F., Asano, H., Murata, H. & Watanabe, N. (1987). *J. Appl. Crystallogr.*, **20**, 411.

Izumi, F. & Takayama-Muromachi, E. (1995). "High-Temperature Superconducting Materials Science and Engineering: New Concepts and Technology," ed. by D. Shi, Pergamon, Oxford, Chap. 3.

Izumi, F. (1996). "Applications of Synchrotron Radiation to Materials Analysis," ed. by H. Saisho and Y. Gohshi and , Elsevier Science, Amsterdam, Chap. 7.

Jandel Scientific (1993). "SigmaPlot, Scientific Graphing Software for Windows, User's Manual."

Johnson, C. K. (1976). "ORTEP-II, a FORTRAN Thermal-Ellipsoid Plot Program for Crystal Structure Illustrations," Report ORNL-5138, Oak Ridge National Laboratory, Oak Ridge, Tennessee.

Kikuchi, M., Izumi, F., Kikuchi, M, Ohshima, E., Morii, Y., Shimojo, Y. & Syono, Y. (1995). *Physica C* (*Amsterdam*), **247**, 183.

Kim, Y.-I. & Izumi, F. (1994). *J. Ceram. Soc. Jpn.*, **102**, 401.

Larson, A. C. & Von Dreele, R. B. (1990). "GSAS, General Structure Analysis System," LAUR 86-748, Los Alamos National Laboratory, Los Alamos, New Mexico.

Malmros, G. & Thomas, J. O. (1977). *J. Appl. Crystallogr.*, **10**, 7.

March, A. (1932). *Z. Kristallogr.*, **81**, 285.

Marciniak, H. (1995). "Plot Program DMPLOT for Viewing Results of DBWS-9006PC Rietveld Analysis Programs."

Morii, Y., Fuchizaki, K., Funahashi, S., Minakawa, N., Shimojo, Y. & Ishida, A. (1992). Proceedings of the 4th International Symposium on Advanced Nuclear Research, p. 280.

Parrish, W. (1992). "International Tables for Crystallography," Vol. C, ed. by A. J. C. Wilson, Kluwer, Dordrecht, pp. 48–50.

Peterse, W. J. A. M. & Palm, J. H. (1966). *Acta Crystallogr.*, **20**, 147.

Post, J. E. & Bish, D. L. (1989). "Modern Powder Diffraction," ed. by D. L. Bish and J. E. Post, Mineral. Soc. Am., Washington, D.C., Chap. 9.

Prince, E. (1981). *J. Appl. Crystallogr.*, **14**, 157.

Prince, E. (1993). "The Rietveld Method," ed. by R. A. Young, Oxford University Press, Oxford, Chap. 3.

Rietveld, H. M. (1967). *Acta Crystallogr.*, **22**, 151.

Rietveld, H. M. (1969). *J. Appl. Crystallogr.*, **2**, 65.

Rotella, F. J. (1988). "Users Manual for Rietveld Analysis of Time-of-Flight Neutron Powder Diffraction Data at IPNS," Argonne National Laboratory, Argonne, Illinois, Chap. 13.

Rouse, K. D., Cooper, M. J., York, E. J. & Chakera, A. (1970). *Acta Crystallogr., Sect. A*, **26**, 682.

Sakthivel, A. & Young, R. A. (1992). "User's Guide to Programs DBWS-9006 and DBWS-9006PC for Rietveld Analysis of X-Ray and Neutron Diffraction Patterns."

Sasa, Y. & Uda, M. (1976). *J. Solid State Chem.*, **18**, 63.

Scott, H.G. (1983). *J. Appl. Crystallogr.*, **16**, 159.

Taylor, J.C. & Mutulus, C.E. (1991). *J. Appl. Crystallogr.*, **24**, 14.

Thompson, P., Cox, D. E. & Hastings, J. B. (1987). *J. Appl. Crystallogr.*, **20**, 79.

Tobochnik, N. & Tobochnik, J. (1993). *Computers Phys.*, **7**, 672.

Toraya, H. & Marumo, F. (1981). *Mineral. J.*, **10**, 211.

Von Dreele, R. B. (1989). "Modern Powder Diffraction," ed. by D. L. Bish and J. E. Post, Mineral. Soc. Am., Washington, D.C., Chap. 11.

Von Dreele, R. B., Jorgensen, J. D. & Windsor, C. G. (1982). *J. Appl. Crystallogr.*, **15**, 581.

WaveMetrics (1994). "Igor Pro, User's Manual," Lake Oswego, Oregon, pp. 193–197.

Wertheim, G. K., Butler, M. A., West, K. W. & Buchanan, D. N. E. (1974). *Rev. Sci. Instrum.*, **45**, 1369.

Wiles, D. B. & Young, R. A. (1981). *J. Appl. Crystallogr.*, **14**, 149.

Young, R. A. (1993). "The Rietveld Method," ed. by R. A. Young, Oxford University Press, Oxford, Chap. 1.

Young, R. A. & Desai, P. (1989). *Arch. Nauk. Mater.*, **10**, 71.

Young, R. A., Sakthivel, A., Moss, T. S. & Paiva-Santos, C. O. (1995). *J. Appl. Crystallogr.*, **28**, 366.

Young, R. A. & Wiles, D. B. (1982). *J. Appl. Crystallogr.*, **15**, 430.