

SEDEM, a Software Package for EXAFS Data Extraction and Modelling

Daniel Aberdam

Laboratoire de Cristallographie, CNRS, 25 Avenue des Martyrs, BP 166, F-38042 Grenoble CEDEX 9, France. E-mail: aberdam@polycnrs-gre.fr

(Received 27 November 1997; accepted 16 March 1998)

A software package for extended X-ray absorption fine structure (EXAFS) data extraction and modelling, running on DOS- or Windows-operated PCs, is described. This package is written with the aim of giving the user a tool to undertake all steps of data processing and modelling, rather than making use of the most recent programming facilities. However, it remains easy to use, and self-explanatory to those who have already worked with EXAFS. It is divided into two main executable pieces of software. The first one is used to extract the EXAFS $k^n\chi(k)$ function from the data and isolate the shell contributions by Fourier filtering. A tool to sum the spectra before EXAFS extraction is provided. The second one is designed to model the EXAFS spectra or the shell contributions, using amplitude and phase data either from McKale's tables, computed from the *FEFF* program or extracted from experimental reference spectra. This modelling program allows either an optimization of the simulation by a least-mean-square gradient algorithm, with a statistical evaluation of the result of optimization, or, in the case of a single shell, a direct determination of the four main parameters (neighbour distance and number, energy shift and Debye–Waller factor) by decorrelation of the phase and amplitude. In the presence of anharmonicity, the cumulant expansion of the radial distribution of distances is obtained from the phase and amplitude decorrelation. This package is in use at the Collaborative Research Group on Interfaces (CRG-IF) bending-magnet #32 X-ray line (BM-32) at the European Synchrotron Radiation Facility (ESRF) in Grenoble, France.

Keywords: EXAFS; data extraction; modelling.

1. Introduction

The reason why it is worth adding yet another piece of EXAFS software to the literature is not the fact that this program is written using up-to-date visual graphic tools, because it is not. In fact, it is written in Turbo Pascal for DOS, but that does not prevent it from running from any version of Windows. It is not because this program is automated so that you have just to use it once, and after that it will process your data during your coffee break, because it is not automated. In fact, it records the steps you made, and you can refer to these previous steps for further processing. It is not because this program is able to simulate all features in spectra, including multiple scattering, because it does not. In fact, it is based on the standard single-scattering EXAFS formula.

The actual reasons for publishing this program are modest, but they are felt to be pertinent to people working in the field.

(i) The program is written in order to give the user the possibility of processing their data with some choice in the tools. The user may control graphically each action they perform, not only the final result, giving them a very accurate control of what they are doing.

(ii) During data processing, the shape of the spectra are standardized according to theoretical calculations, which improves greatly the significance of comparing spectra. To this end the operation of baseline determination (the atomic absorption) is split into two steps, one in energy space and the other in k -space. As a by-product, this makes the determination of the atomic absorption easier.

(iii) At the stage of computing the Fourier transform, in view of filtering the shell contributions *via* back Fourier transform, a provision is made to evaluate the noise on the processed data, information required to estimate uncertainties on adjusted parameters, and to compute a quality factor.

(iv) In order to model the data, amplitude and phase tables must be available. The present software makes it possible to use either McKale's tables (McKale *et al.*, 1988), tables computed by the *FEFF* program (Rehr *et al.*, 1992), or tables previously extracted (using this software) from reference data.

(v) In the case where a single-shell contribution is modelled, and where the parameters associated with amplitude and phase tables are known (tables obtained from *FEFF* or from extraction of reference data), then the software makes it possible to find the shell parameters by

direct decorrelation of the phase and amplitude (Teo, 1986) instead of performing the minimization of some distance function, such as a least-mean-square distance. This is a very powerful method which, in addition, permits one to account for non-harmonic distribution of distances by means of a cumulant expansion (Dalba *et al.*, 1993).

(vi) When a cumulant expansion is found necessary, it is possible to build back the actual distribution of distances from that cumulant expansion.

(vii) In other cases, a Marquardt-type gradient algorithm (Marquardt, 1963) performs the minimization of the least-mean-square distance between the model and Fourier-filtered data, or between the model and the raw $k^n\chi(k)$ function.

(viii) Uncertainties in results are evaluated in two steps. The first step is the calculation of the standard deviation for individual uncorrelated parameters by use of the covariance matrix. The second step is the research of the effect of correlations by investigating correlations between all relevant pairs of parameters.

The software is split into two main executable units, the first, *xafs.exe*, to prepare the data from the raw spectra, the second, *modl.exe*, to model the data. In addition, a small unit, *xafsSum.exe*, is used to sum the spectra before processing, with special care given to the dispersion of data in energy. A second small unit, *cumul.exe*, performs the building of the distribution of distances from the cumulant expansion.

This paper is divided into four parts. §1 deals with the description of data processing (*xafs.exe*), §2 with data modelling (*modl.exe*), §3 enters into some details of the evaluation of uncertainties, and §4 shows the capabilities of the software in an example dealing with the semiconductor-to-metal transition of selenium close to its critical point.

2. Data processing: *xafs.exe*

It is frequent that, with difficult samples, the general shape of an EXAFS spectrum is distorted with respect to the ideal shape because of the presence of undesirable photons at the detector window when detecting fluorescence. This distortion prevents a comparison with other spectra from being meaningful, and makes it more difficult to extract the EXAFS data. The *xafs.exe* program has been written with the aim of minimizing these drawbacks. To this end, it is divided into three parts. The first part works on the data in energy space, to obtain the spectrum $\mu(e)$, normalized in shape. The second part works on data in momentum space to obtain $k^n\chi(k)$, and the third part works in real space, to obtain the Fourier transform of $k^n\chi(k)$ and filtered back Fourier transforms. Prior to any processing, the user may sum the raw spectra using *xafsSum.exe*, which takes care of a possible dispersion of the abscissas (energy) values.

2.1. Processing in energy space

Processing in energy space is designed to obtain a spectrum normalized in shape, despite the distortions it

suffers from. Whatever the data collection mode, this shape is chosen as that of the theoretical energy dependence of the atomic absorption, $\mu_0(e)$, relative to its value, h_s , at the threshold energy, e_s . This shape is computed internally by means of a six-parameter polynomial fit to the μ/ρ output of the *f-Prime* program of Cromer & Liberman (1970). This polynomial fit is very accurate, and does not differ from the original μ/ρ by more than the inaccuracies still present in the *f-Prime* computation, due to residual approximations.

In the case where this procedure fails for any reason, the atomic absorption is calculated with a formula quantum-mechanically derived by Stobbe (1930), from the K orbital functions of an H atom. The Stobbe function, quoted by James (1950) for K -shell ionization, is the following,

$$\mu_0(e) = \mu_0(e_s) \exp\{4[1 - \arctan(x^{1/2})/x^{1/2}]\}/(1+x)^4 \times [1 - \exp(-2\pi/x^{1/2})], \quad (1)$$

where $x = (e - e_s)/e_s$. Stobbe also calculated explicit formulae for L_1 - and L_{23} -shell ionization. However, only the formula for K -shell ionization is probably a sufficiently good approximation, according to James. The Stobbe formulae for K - and L -shell ionizations have been compared with McMaster's tables (McMaster *et al.*, 1969), Sasaki's tables (Sasaki, 1989), Cromer's *f-Prime* program output and the following power law,

$$\mu_0(e) = \mu_0(e_s) (e_s/e)^3. \quad (2)$$

Keeping the μ/ρ output of the *f-Prime* program as the basis for the comparison, it turns out that the differences between Sasaki's tables and the *f-Prime* output are always marginal for EXAFS analysis purpose. McMaster's tables are crudely sampled, but they could be convenient for EXAFS analysis. The Stobbe formula for K -shell ionization is often close to the *f-Prime* output, whatever the actual

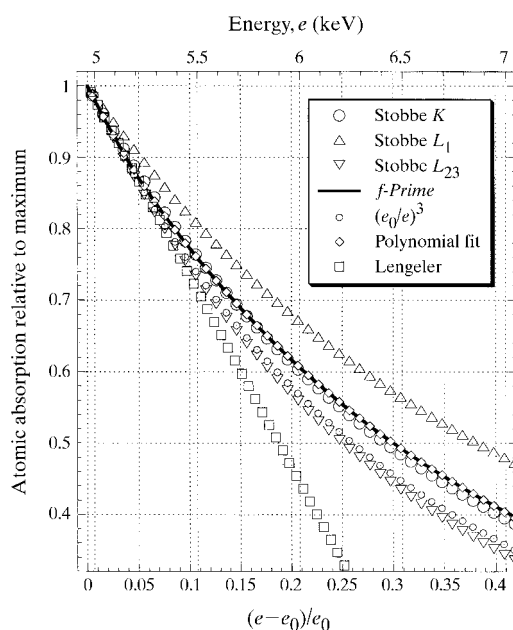


Figure 1 Atomic absorption at the Ti K edge. $e_0 = 4.966$ keV.

edge under examination. Apart from some exceptions, the Stobbe formula is generally much closer to the *f-Prime* output than the power law. However, the latter looks slightly better for some *K* edges in the 10–20 keV range. The Lengeler normalization (Lengeler & Eisenberger, 1980) is a straight-line tangent to the Stobbe function at $e = e_s$. Owing to the closed form of the Stobbe formula, there is no advantage in replacing it by a straight line, and it will be used as the atomic normalization function in this program in replacement of the polynomial fit procedure in case of failure of the latter.

Figs. 1 and 2 show the atomic absorption relative to its maximum as computed by the three Stobbe formulae, the *f-Prime* program, the power law, the polynomial fit designed in *SEDEM*, and the Lengeler approximation, in the cases of the titanium *K* edge and the platinum L_3 edge. The plots of the polynomial fit and *f-Prime* are virtually identical. It may also be observed in Fig. 2 that the Stobbe formula for *K*-shell electrons is very close to the *f-Prime* output, even for *L* shells, as in the case of platinum.

The parameters required to compute the parameterized fit are stored in a text file included in the software package. The necessary steps of this normalization to atomic absorption are the determination of the threshold energy, e_s , and of the edge height, h_s , at the edge energy. Before this, cleaning tools are available, if required.

2.1.1. *Cleaning tools.* After reading the raw data, two menu items are offered to first correct the data from outlying points or glitches. (i) Outliers are easily suppressed by replacing their ordinate by the mean value of the ordinates of the two pairs of adjacent points. A graphic selection plus a single key-stroke is enough to do the job. (ii) The program offers the facility of reading and displaying I_0 , the beam intensity before absorption, together with the spectrum, to help localize glitches. Two methods of deglitching

are offered. The first one performs a high-pass FFT filtering within the selected range; the second one uses a cubic-spline smoothing algorithm, using weights smaller within the glitch range than outside. Both methods may give good or bad results. The narrower the glitch, the better the result, of course.

2.1.2. *Threshold energy and height of the edge.* The threshold energy, e_s , is chosen graphically. The program proposes a choice at the maximum in the derivative of the spectrum with respect to energy. The computation of this derivative has been given special attention to avoid both noise and distortion. It must be stressed that this determination of the threshold energy has no physical meaning by itself. It must be subjected to a critical examination, as illustrated by Krill (1986).

The height of the edge, h_s , is determined at the threshold energy, e_s . The bottom point is obtained by right-hand extrapolation over a range of a few eV of a straight line fitted by least-mean-square (lms) methods to the pre-edge part of the spectrum. The top point is also obtained by left-hand extrapolation over a range of a few eV of a baseline simulating the atomic absorption, going through the EXAFS oscillations. Three tools are provided to build this baseline: a polynomial lms fitting, a gliding-window smoothing, and a cubic-spline smoothing algorithm. The latter, using a variety of weight distributions, is the most powerful. It is very important at this stage to carefully avoid oscillations in this baseline. On the other hand, it must pass through the oscillations in a well balanced manner. A good baseline is essential for the next step, the normalization of the shape of the spectrum using the *f-Prime* program or the Stobbe formula [equation (1)].

2.1.3. *Shape normalization using the f-Prime program or the Stobbe formula.* This step is very simple and needs no input from the user. It simply consists of replacing the baseline determined in the previous step by the theoretical atomic absorption obtained from the parameterization of the μ/ρ output of the *f-Prime* program or by the Stobbe formula [equation (1)]. At this stage, the spectrum may be recorded in ASCII format (two columns), either as it is or normalized to the edge height. This makes the comparison with other spectra meaningful and easy. This step closes the chapter of processing in energy space.

2.2. Processing in momentum space

After shape normalization in energy space, processing in momentum space includes the conversion of the spectrum to the momentum scale, a refinement of the determination of the baseline simulating the atomic absorption, the subtraction of the baseline from the oscillations and the normalization, ending with the $\chi(k)$ function. This function may then be weighed by k^n , n being 0, 1, 2 or 3. Apodization then prepares the $k^n\chi(k)$ function to be Fourier transformed. The various steps described here may differ technically from other software packages. However, they are rather 'classical', so that the discussions about how or how not to treat the various parameters (e.g. in baseline refining,

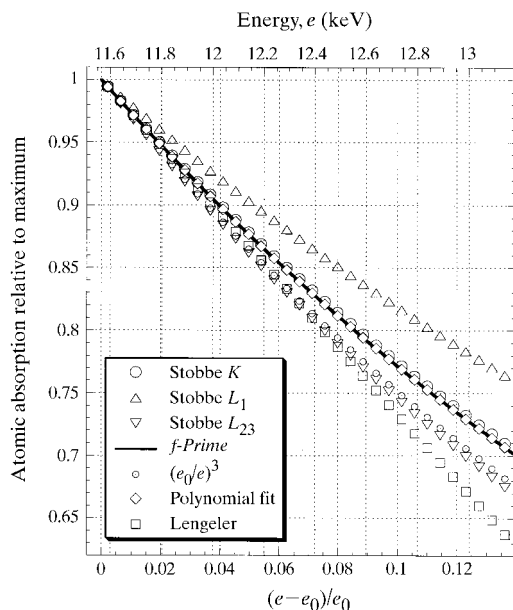


Figure 2
Atomic absorption at the Pt L_3 edge. $e_0 = 11.656$ keV.

in weighing, in apodizing, in Fourier transforming forth and back) are still valid. Such a discussion has been carefully compiled by Michalowicz (1990).

2.2.1. *Baseline refinement.* $\mu_0(e)$ is the theoretical function which describes atomic absorption in energy space. $\mu_0(k)$ is the corresponding function in momentum space. The actual baseline, which minimizes low-‘frequency’ amplitudes (low means ‘frequencies’ smaller than e.g. 1.5 Å) in the Fourier transform of $k^n\chi(k)$, has still to be refined, starting from the (not so bad) $\mu_0(k)$ function, already determined in energy space. Four tools are provided to this end, with a display of the derivative of the baseline, to help avoid undesirable oscillations, a very important requirement: a low-pass FFT filtering, a gliding-window algorithm, an lms polynomial fitting including positive and negative powers of k , and a weighed cubic-spline smoothing algorithm, with various choices of weights.

2.2.2. *Baseline subtraction and normalization, k^n weighing and apodization.* Once the baseline is considered satisfactory, it is subtracted from the spectrum $\mu(k)$ and normalization is performed, according to

$$\chi(k) = [\mu(k) - \mu_0(k)]/\mu_0(k). \quad (3)$$

According to the problem under study, $\chi(k)$ may be weighed by a power function of k , with integer exponent from 0 to 3, leading to $k^n\chi(k)$. Apodization is aimed to minimize truncation effects when Fourier transforming $k^n\chi(k)$. The windows recommended are either the normal or the plateau Kaiser windows.

2.3. Processing in real space

Processing in real space consists of Fourier transforming the function $k^n\chi(k)$, in view of gaining a global view of the raw distance distribution (‘raw’ means uncorrected for phase shifts and amplitude functions). The Fourier transform and its inverse are computed directly, as a Fourier integral, which is more versatile than the fast Fourier transform algorithm. The inverse Fourier transform is renormalized by the apodization window which was used to prepare the Fourier transform. It may happen that the Fourier transform shows a low-frequency unphysical peak which overlaps with the peak corresponding to the first-neighbour distance. In such a case, care must be taken to avoid distortion of the physical peak by overdamping the unphysical peak. This may require returning to the baseline subtraction.

The maximum value of the distance, r_{\max} , up to which the Fourier integral may be computed, is defined according to the Nyquist or Shannon criterion, that is 2π times the reverse of the largest momentum step Δk_{\max} in $k^n\chi(k)$,

$$r_{\max} = \pi/\Delta k_{\max}. \quad (4)$$

If r_{\max} is large enough in comparison with the distance of the outermost shell contributing to the spectrum, r_{outer} , then the amplitudes located between r_{outer} and r_{\max} may be interpreted as high-frequency noise amplitudes. The back

Fourier transform of this part of the spectrum gives an estimation of the noise in momentum space. This information, which is difficult to obtain directly from the measurements because of the tricky nature of the data processing, is very useful because it makes it possible to evaluate uncertainties in the parameter determinations. This question of evaluating uncertainties will be addressed in more detail in §4.

3. Data modelling: *modl.exe*

This piece of software is designed to perform various tasks, which require the use of common tools, so it may appear slightly intricate. However, its structure is legible.

(i) Simulation: entering a set of parameters for one or more shells (up to five), reading the necessary phase and amplitude files, the software computes the modelled $k^n\chi(k)$ spectrum and its Fourier transform.

(ii) Extraction of phase and amplitude functions from reference data: the extracted functions are absolute functions, renormalized with respect to the structural shell parameters, and the pre-factor.

(iii) Determination of shell parameters by an lms minimization of the ‘distance’ between the simulation, on the one hand, and ‘data’, that is either the measured $k^n\chi(k)$ spectrum or the Fourier filtered shell contribution, on the other. McKale’s tables, *FEFF* calculations or tables extracted from reference data may be used for the simulation.

(iv) In the case where a reference spectrum is available (either from an experiment or a calculation), and the parameters r and N for phase and amplitude extraction are known, a shell-parameter determination alternative to the lms minimization is possible for a single shell. This method uses decorrelation of phases and amplitudes.

(v) Cumulant expansion for non-Gaussian radial distribution function of distances.

(vi) Recovery of the radial distribution function of distances from cumulant expansion: the *cumul.exe* unit.

(vii) Calculation of the Fourier transforms of the $k^n\chi(k)$ spectrum and of the optimized simulation, and of their difference. Fourier transforms are displayed together for comparison.

3.1. Simulation

Simulation is based on the standard EXAFS formula. It is very useful to rapidly check whether a structural assumption is reasonable or not by comparing its Fourier transform with that of the experimental or reference spectra. Tables of phase and amplitude (McKale’s, *FEFF* or extracted from reference data) are cubic-spline interpolated on the simulation grid. The first and last values of that grid are between 1.5 and 25 Å⁻¹. Shell parameters may be either read from a file or entered *via* the keyboard. A parameter file contains as many blocks as the number of shells involved. The blocks are preceded by the number of shells, and followed by the value of the pre-factor S_0^2 . Each

block contains the atomic number of the scattering atom, the interatomic distance r , the number of neighbours N , the energy shift Δe_0 , the Debye–Waller factor parameter σ^2 , plus three parameters, γ , η , β , describing the inelastic mean free path λ of electrons, according to the model

$$\lambda = [(\eta/k)^4 + k + \beta k^2]/\gamma. \quad (5)$$

This model is a modified version of that used by Michalowitz (1990). In his work, the βk^2 term is not present so that λ is linear in k at large values of k . However, it turns out that inelastic mean free paths computed by the *FEFF* program are not linear but show a parabolic dependence on k for large k . That is the reason why the parameter β has been introduced. If tables are read from an *FEFF* file then the software determines automatically the three parameters defining λ by least-mean-square fitting of the *FEFF* table using equation (5). Otherwise, β is generally assumed to be zero.

Once the simulation is computed, the same apodization windows as in *xafs.exe* are offered to compute its Fourier transform.

3.2. Extraction of phase and amplitude tables from reference data

This can be performed only for one shell at a time. In principle, extraction of absolute amplitude and phase are possible, given the structural parameters r , N and σ^2 , plus λ and S_0^2 . However, it often happens that σ^2 , λ and S_0^2 are not well known. It is then useful to set these parameters somewhat arbitrarily by performing a simulation of the data with this set of parameters and by optimizing the simulation, keeping fixed the known structural parameters. In output, the user is given optimum values of σ^2 , λ and S_0^2 , which must be looked at critically. These parameters must be given sensible values to extract meaningful phase and amplitude tables. As a rule they might be kept fixed in further analysis of an unknown sample.

3.3. Determination of shell parameters by distance minimization

If the conditions required to decorrelate the phase and amplitude are not fulfilled (*i.e.* one shell at a time, and phase and amplitude tables with known r and N extraction parameters), then one may use a minimization algorithm to find out the shell parameters. The algorithm is a modified Marquardt (1963) algorithm. The distance to be minimized is the sum of the square of the differences between the data and simulation, normalized by the standard deviation σ_i at each point of the data set,

$$X^2 = \sum_{i=1}^{n_{\text{data}}} \{[y_i - y(x_i, r, N, \sigma_{\text{dw}}^2, \lambda)]/\sigma_i\}^2. \quad (6)$$

This calculation requires knowledge of the σ_i values. How this knowledge is obtained will be explained in the discussion below. The algorithm makes use of the derivatives of the EXAFS function with respect to the parameters. With these derivatives it computes correlation and

covariance matrices, from which uncorrelated standard deviations, quality factor, and two-parameter correlations are deduced. The number of independent data points, n_{indep} , is determined from the width of the filtering window in real space, Δr , times the width of the apodization window in momentum space. According to Stern (1993), this number is the integer closest to

$$n_{\text{indep}} = (2\Delta r \Delta k_{\text{apod}}/\pi) + 2. \quad (7)$$

The maximum number of adjustable parameters, $n_{\text{fit}}^{\text{max}}$, is

$$n_{\text{fit}}^{\text{max}} = n_{\text{indep}} - 1. \quad (8)$$

3.4. Determination of shell parameters by phase and amplitude decorrelation

Whenever possible (*i.e.* when one shell only is studied, and a ‘good’ reference spectrum is available, accompanied by known values of the parameters used to perform the extraction of phase and amplitude), it is recommended to use the phase and amplitude decorrelation method to determine the shell parameters. The principle of the method is described by Teo (1986). A ‘good’ table of phase means that the phase difference between the reference and sample mainly arises from a difference in neighbour distance. Phase shifts due to atom scattering properties are assumed to cancel (transferability). Thus the phase difference should be linear in k and extrapolate to zero for zero momentum,

$$\Delta\varphi = 2k(r_{\text{sample}} - r_{\text{ref}}). \quad (9)$$

If this is not the case, it means that the value of the energy shift Δe_0 for the sample with respect to the reference is not good. One must then find a value of Δe_0 such that a straight line, least-square fitted on $\Delta\varphi$, extrapolates to zero for zero k . The unknown neighbour distance r_{sample} is then derived from the slope p of that straight line: $r_{\text{sample}} = p/2 + r_{\text{ref}}$. Similarly, the logarithm of the ratio of amplitudes is linear in k^2 , provided the difference $2[r_{\text{ref}}/\lambda_{\text{ref}}(k) - r_{\text{sample}}/\lambda_{\text{sample}}(k)]$ may be neglected in the whole momentum range, which is usually a reasonable approximation. One obtains

$$\ln(A_{\text{sample}}/A_{\text{ref}}) \simeq \ln[(N_{\text{sample}}/N_{\text{ref}})(r_{\text{ref}}^2/r_{\text{sample}}^2)] + 2k^2(\sigma_{\text{ref}}^2 - \sigma_{\text{sample}}^2). \quad (10)$$

From the slope of a straight line least-square-fitted on $\ln(A_{\text{sample}}/A_{\text{ref}})$, σ_{sample}^2 may be derived, and, from its null momentum intercept, one obtains N_{sample} .

This method is very powerful because it unravels strong correlations between the distance and energy shift on the one hand, and the number of neighbours and Debye–Waller parameter on the other. However, it works properly only if the phase difference and amplitude ratio behave in a sufficiently regular manner, so that the least-mean-square linear or polynomial fits provide meaningful parameters (slopes, zero intercept, cumulants).

3.5. The cumulant expansion

In the standard single-scattering approximation, the EXAFS analysis does not determine the actual radial distribution function (RDF), $\rho(r)$, but a so-called effective distribution,

$$P(r, \lambda) = \rho(r)[\exp(-2r/\lambda)/r^2]. \quad (11)$$

If $\rho(r)$ is not Gaussian or not narrow enough, the difference between the actual and effective distribution function is not negligible, and the use of the standard formula may lead to significant errors in the parameter determinations. It is then useful to expand, in the vicinity of null momentum, the Fourier transform of the effective radial distribution function, which gives the EXAFS oscillations, into a series of cumulants, C_n ,

$$\int_0^{\infty} P(r, \lambda) \exp(2ikr) dr = \exp\left(\sum_{n=0}^{\infty} \{[(2ik)^n/n!]C_n\}\right). \quad (12)$$

C_0 depends on the normalization of the distribution, C_1 and C_2 correspond, respectively, to the mean value of the interatomic distance r and its variance σ^2 , averaged over the effective RDF. Higher-order cumulants characterize the shape of the distribution. They are zero for a Gaussian distribution. It is possible to express the EXAFS function in terms of cumulants,

$$k\chi(k) = A(k) \sin[\varphi(k)], \quad (13)$$

$$\varphi(k) = 2C_1k - \frac{4}{3}C_3k^3 + \frac{8}{15}C_5k^5 - \dots + \dots, \quad (14)$$

$$A(k) = S_0^2 N |f(k, \pi)| \exp(C_0 - 2C_2k^2 + \frac{2}{3}C_4k^4 - \dots + \dots). \quad (15)$$

It follows that the phase difference and the logarithm of the amplitude ratio are given by

$$\Delta\varphi(k) = 2(\Delta C_1)k - \frac{4}{3}\Delta C_3k^3 + \frac{8}{15}\Delta C_5k^5 - \dots + \dots, \quad (16)$$

$$\ln\left[\frac{A_{\text{sample}}(k)}{A_{\text{ref}}(k)}\right] \simeq \ln\left[\frac{N_{\text{sample}}}{N_{\text{ref}}}\right] + \Delta C_0 - 2\Delta C_2k^2 + \frac{2}{3}\Delta C_4k^4 - \dots + \dots \quad (17)$$

The values C_n and the coordination number N are obtained by following a procedure similar to the one used for phase and amplitude decorrelation, described in §3.4, by replacing the straight-line least-mean-square fitting of the phase difference and of the logarithm of amplitude ratio by a polynomial least-mean-square fitting. The higher the degree of the polynomial, the larger the number of cumulants in the expansion. It must be stressed here that the number of cumulants required to describe a non-Gaussian distribution of distances is not easy to determine. It must be large enough to obtain a convergence when rebuilding the

radial distribution function, and small enough to keep reasonable values and lead to coherent results all along the analysis. This point is illustrated below in the case of selenium, close to its critical point.

3.6. The radial distribution function: *cumul.exe*

The *cumul.exe* unit restores the actual radial distribution function of distances from the cumulant expansion determined in the phase and amplitude decorrelation step. This is possible only if the cumulant expansion converges in the momentum range of interest. This depends on the shape of $\rho(r)$ and $P(r, \lambda)$ and on the number of cumulants used to build the expansion. The convergence is checked by the program. The formalism of cumulant expansion is built assuming a constant value of λ , whereas this quantity is generally assumed to be k -dependent. To solve this difficulty, λ is averaged over the relevant momentum range, and it may be verified that the resulting RDF is not very sensitive to the actual value of λ . The program computes the right-hand side of equation (12); then it computes its inverse Fourier transform, obtaining $P(r, \lambda)$, from which $\rho(r)$ is obtained using equation (11). The asymmetry of $\rho(r)$ is measured as the difference between the right and left half-widths at half-maximum normalized to full width at half-maximum.

4. Evaluation of uncertainties

The evaluation of uncertainties is a rather tricky problem in EXAFS. This discussion is organized as a question and answer dialogue, modelled on a questionnaire issued from the Committee of Standards and Criteria of the International XAFS Society with the aim of proposing a standardized method for reporting error estimates in EXAFS results. Most of the following definitions and solutions are taken from *Numerical Recipes* (Press *et al.*, 1986).

4.1. Form of function being minimized

In the *SEDEM* package, EXAFS parameters are not always determined by minimizing a 'distance' between a raw spectrum or a filtered shell contribution, on the one hand, and a model simulation on the other. When a single shell is to be studied and 'good' references are available, then parameters are determined directly by phase and amplitude decorrelation. In other cases, a Marquardt-type gradient algorithm is used to minimize the distance between the data and model. The distance function being minimized is a mean-square distance weighed with the statistical uncertainties, equation (6). In that equation, the table of σ_i values is needed. Various ways of building this table are explained in the following.

4.2. Provision for estimating experimental errors

The purely statistical uncertainty, which is proportional to the square root of the number of counts, is difficult to propagate through the analysis procedure. Thus, in the *SEDEM* package, the table σ_i of statistical uncertainties is

not determined rigorously but is estimated using one of a few different possibilities.

4.2.1. *Back Fourier transform of high-frequency noise in real space.* This first possibility works well whatever the number of shells to simulate, but it requires that the maximum acquisition step in momentum space is small enough, so that the maximum significant value of distance in real space, r_{\max} (imposed by Shannon's theorem), up to which the Fourier transform may be computed, is significantly larger than the maximum distance, r_{outer} , where a shell contribution is detectable [equation (4)]. Within this condition, the back Fourier transform of that part of the spectrum comprised between r_{outer} and r_{\max} may be considered as an estimation of the noise in momentum space.

Taking a smoothed shape of the amplitude of that noise gives a table of k -dependent σ_i values. Smoothing is mandatory, because small noise amplitudes will result in heavily weighed points distributed at random, making minimization meaningless. However, if the σ_i values are k -dependent, the standard deviations for uncorrelated parameters, computed from the covariance matrix (see below), depend on the actual shape of $\sigma_i(k)$. On the contrary, if the σ_i values are not k -dependent (*i.e.* if all σ_i are equal to σ), then the standard deviations for uncorrelated parameters are not dependent on the value of σ . Consequently, it is recommended, but not mandatory, to replace the σ_i values by their root-mean-square values σ .

4.2.2. *Distance between filtered back Fourier transform and EXAFS spectrum.* This second possibility of estimating the statistic uncertainty table σ_i may be used when the previous one does not work, *i.e.* when r_{\max} is not large enough with respect to r_{outer} . It works under the restriction that the Fourier-filtered signal includes the totality of the shells which contribute significantly to the signal. Within this condition, the estimation of the noise amplitude is computed as the difference between the raw EXAFS and the Fourier-filtered EXAFS,

$$\sigma_i^2 = (y_{\text{filtered}} - y_{\text{exp}})_i^2. \quad (18)$$

Here again it is recommended to choose the common value σ of noise amplitude as the root mean square of σ_i , but it is also possible to obtain a momentum-dependent table of σ_i .

These two ways of estimating the uncertainty table are not based on a rigorous reasoning, but they behave in a rather reasonable manner. In addition, they are obtained with the help of simple and reproducible recipes, not very sensitive to user's fingers.

Systematic errors in tables of phase and amplitude, either computed *ab initio* or extracted from experimental references, are not taken into account in this software. Systematic errors due to baseline subtraction and renormalization are accounted for implicitly when using the second method of estimation of the σ_i table just described. The relationship between the uncertainty table in momentum space and its real-space counterpart is clear in the case of the first estimation method described before. It

is a Fourier transform relationship. However, no specific table of uncertainties is built by this program in real space.

4.3. Methods used to calculate confidence limits

In this software, confidence limits are calculated in two steps: standard deviations are first computed for parameters with correlations excluded, then confidence limits are set after two-parameter correlations are examined. The mathematics is taken entirely from *Numerical Recipes* (Press *et al.*, 1986).

4.3.1. *Standard deviations for parameters with correlations excluded.* The standard deviation attached to parameter par_j is proportional to the square root of the corresponding diagonal element of the covariance matrix, $\text{cov}[j,j]$, the definition of which is given later,

$$\sigma(\text{par}_j) = (\delta X^2)^{1/2} (\text{cov}[j,j])^{1/2}. \quad (19)$$

The coefficient δX^2 is a function of the confidence level and of the number of free parameters, n_{fit} . In the present case, as we calculate the standard deviation for each parameter taken separately, then $n_{\text{fit}} = 1$. If we choose a confidence level $\text{cl} = 0.683$ (a value related to the normal distribution law), then $\delta X^2 = 1$ and (19) reduces to

$$\sigma(\text{par}_j) = (\text{cov}[j,j])^{1/2}. \quad (20)$$

Let us now recall an important remark: if the statistical uncertainty table σ_i is not momentum dependent, then the standard deviations found at this stage do not depend on the common value σ of σ_i . On the contrary, if σ_i is momentum dependent, then the standard deviations found at this stage do depend on the σ_i table. A second important remark is that the standard deviation derived previously must not be given an actual statistical meaning, because the table σ_i has no clear relationship with a *normal distribution* of errors.

A given parameter may be set adjustable, or fixed, or linearly linked to another parameter. The actual number of

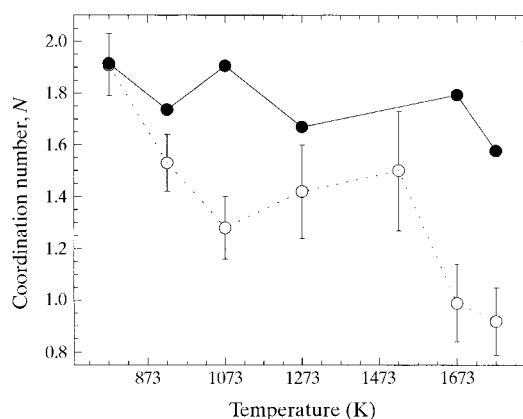


Figure 3 Coordination number *versus* temperature with harmonic (empty circles, dotted line) and anharmonic (filled circles, full line) analyses.

adjustable parameters must be less than or equal to $n_{\text{fit}}^{\text{max}}$ [equation (8), §3.3].

The curvature matrix α is computed from the derivatives of the EXAFS function with respect to the adjustable parameters, after minimization is performed,

$$\alpha[j, k] = \sum_{i=1}^{n_{\text{data}}} (\text{der}[j, i] \text{der}[k, i] / \sigma_i^2), \quad (21)$$

where i runs on the momentum grid, j runs from 1 to the number of free parameters n_{fit} , and k runs from 1 to j . $\text{der}[j, i]$ and $\text{der}[k, i]$ are the derivatives of the EXAFS function with respect to parameters j and k , respectively. The correlation matrix is obtained from the curvature matrix by

$$\text{correl}[j, k] = \alpha[j, k] / [X^2(n_{\text{data}} - n_{\text{fit}})], \quad (22)$$

where X^2 is given by equation (6).

The normalized curvature matrix

$$\alpha_{\text{red}}[j, k] = \alpha[j, k] \{(\alpha[j, j])^{1/2} (\alpha[k, k])^{1/2}\} \quad (23)$$

is inverted to obtain the covariance matrix $\text{cov}[j, k]$. Global correlation coefficients are deduced from the diagonal elements of the covariance and correlation matrices,

$$\text{coef}_{\text{corr}}^{\text{global}}[j] = \{1 - 1/(\text{cov}[j, j] \text{correl}[j, j])\}^{1/2}. \quad (24)$$

In addition, the program computes a quality factor of the minimization, qf , which depends on the number of degrees of freedom, the value of X^2 , the number of data and of free parameters. qf is a number comprised between 0 and 1. It is defined as the solution of

$$qf = \Gamma(n_{\text{freedom}}/2, X_{\text{norm}}^2/2), \quad (25)$$

where Γ stands for the complement of the incomplete gamma function, $n_{\text{freedom}} = n_{\text{fit}}^{\text{max}} + 1 - n_{\text{fit}} = n_{\text{indep}} - n_{\text{fit}}$, and $X_{\text{norm}}^2 = X^2 n_{\text{freedom}} / (n_{\text{data}} - n_{\text{fit}})$. If the common value σ of the elements of the table of σ_i obtained from one of the procedures described in §4.2.1 or §4.2.2 is large, meaning ‘noisy’ measurements, the weight of data points is weak, and whatever the optimization will be, the quality factor will be close to unity and the fit good, in the opinion of the computer. On the contrary, if σ is small, the weight of data is large, and a very good fit will be required to obtain a reasonably high quality factor. If σ_i are momentum dependent, then they generally increase with increasing momentum because they are obtained from data weighed by k^n [this does not imply, of course, that the actual (but often unknown) standard deviations on measurements increases with energy]. The weight of large- k data thus decreases, which sometimes prevents the minimization algorithm from finding an acceptable solution, whereas the quality factor is close to unity.

4.3.2. Two-parameter correlations. This second step consists of calculating correlations of each parameter par_j with every other parameter, par_k , and to select the largest of the uncertainty domains found. The uncertainty domain for par_j in the pair $\text{par}_j, \text{par}_k$ is the width of a contour of constant δX^2 , measured along the axis relative to the

parameter of interest, the value of δX^2 depending on the value of the confidence level (chosen as $\text{cl} = 0.683$ in the program), and on the number of free parameters (*i.e.* $n_{\text{fit}} = 2$ in the present context). δX^2 is the solution of the equation

$$\Gamma(n_{\text{fit}}/2, \delta X^2/2) = 1 - \text{cl}. \quad (26)$$

The solution is $\delta X^2 = 2.30$ for these values of n_{fit} and cl . Once the value of δX^2 is determined, the contour may be either computed from the covariance matrix (faster), or computed on a grid (useful if the actual shape deviates much from an ellipse). The ellipse equation is

$$\delta X^2 = \alpha[j, j] \sigma(\text{par}_j)^2 + 2\alpha[k, j] \sigma(\text{par}_j) \sigma(\text{par}_k) + \alpha[k, k] \sigma(\text{par}_k)^2, \quad (27)$$

where $\alpha[k, j]$ is an element of the curvature matrix.

The calculation of the contour on a grid is performed by giving the two parameters a set of increments, then calculating the ‘distance’ between the data and the model, and finally interpolating linearly within the two-dimensional array of values for distance = δX^2 .

No correction to these estimations of uncertainties for inadequate estimates of experimental errors is made.

5. Example: semiconductor-to-metal transition of selenium

Some work has already been accomplished with the help of this software. For example, (i) the analysis of the semiconductor-to-metal transition of fluid selenium close to its critical point, which made use of the cumulant expansion (Soldo *et al.*, 1998); (ii) the study of the ferric uptake regulation protein isolated from *Escherichia coli*, which provides the first unambiguous evidence for the presence of a structural zinc site in that protein (Coy *et al.*, 1994; Jacquamet *et al.*, 1998); (iii) others, already published or in preparation.

Let us illustrate the capabilities of *SEDEM* in the case of selenium, the analysis of which has required the use of cumulants. In the semiconductor phase of trigonal crystal-

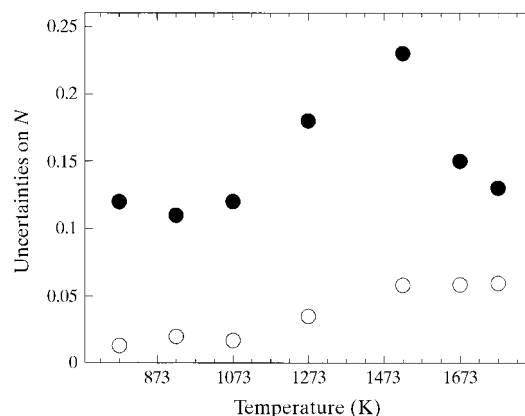


Figure 4 Uncertainties for N , from covariance matrix only, and including pair correlations. Filled circles: total uncertainties; empty circles: uncertainties without correlations.

line selenium, stable in normal conditions, Se atoms are arranged in helical chains; the bond length is 2.33 Å and the bond angle is 103°. Assuming that there are no interchain bonds, the coordination number N of each Se atom is 2 except at the chain ends, and is related to the chain length n by the formula $N = 2 - 2/n$. The length of the chains in normal conditions is of the order of $n = 10^4$ atoms, and

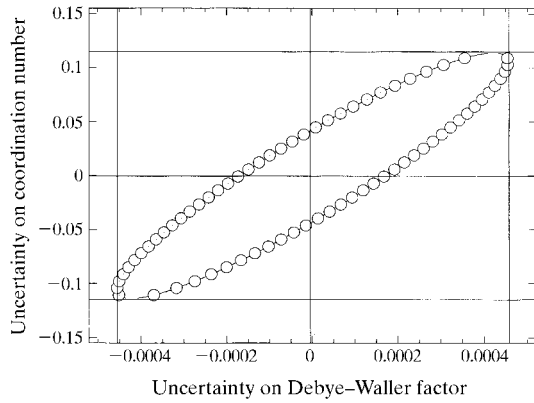


Figure 5

Constant probability contour for N and σ^2 . For the coordination number, $\delta N = 0.021$ ignoring correlations, and $\delta N = 0.11$ taking them into account.

decreases to a few atoms with increasing temperature and pressure in the liquid semiconductor phase.

Close to its critical conditions (1893 K, 385 bar), selenium undergoes a semiconductor-to-metal transition, the mechanism of which is not well known, mainly due to the lack of accurate structural information. New EXAFS measurements have been performed close to the critical point, at a pressure of 600 bar, and a variable temperature, in order to cross the line of the semiconductor-to-metal transition. The experimental conditions and the analysis are described in detail by Soldo *et al.* (1997), where references to previous works may also be found.

Reference phases and amplitudes were extracted from a reference spectrum recorded at 773 K and 100 bar. Parameters for the reference spectrum were $r = 2.33$ Å, $N = 2$, $\sigma^2 = 9.216 \times 10^{-3}$ Å², $S_0^2 = 1$. The electron mean free path was described by $\lambda(k) = k/\gamma$, where γ was kept constant, whatever the temperature and pressure. With extracted phases and amplitudes, the standard minimization procedure offered by *SEDEM* was used to determine the structural parameters *versus* temperature, within the harmonic approximation. In the following, we focus on results relative to the coordination number.

Fig. 3 displays the results of the analysis in the harmonic approximation (*i.e.* with cumulants $C_1 = r$ and $C_2 = \sigma^2$), and

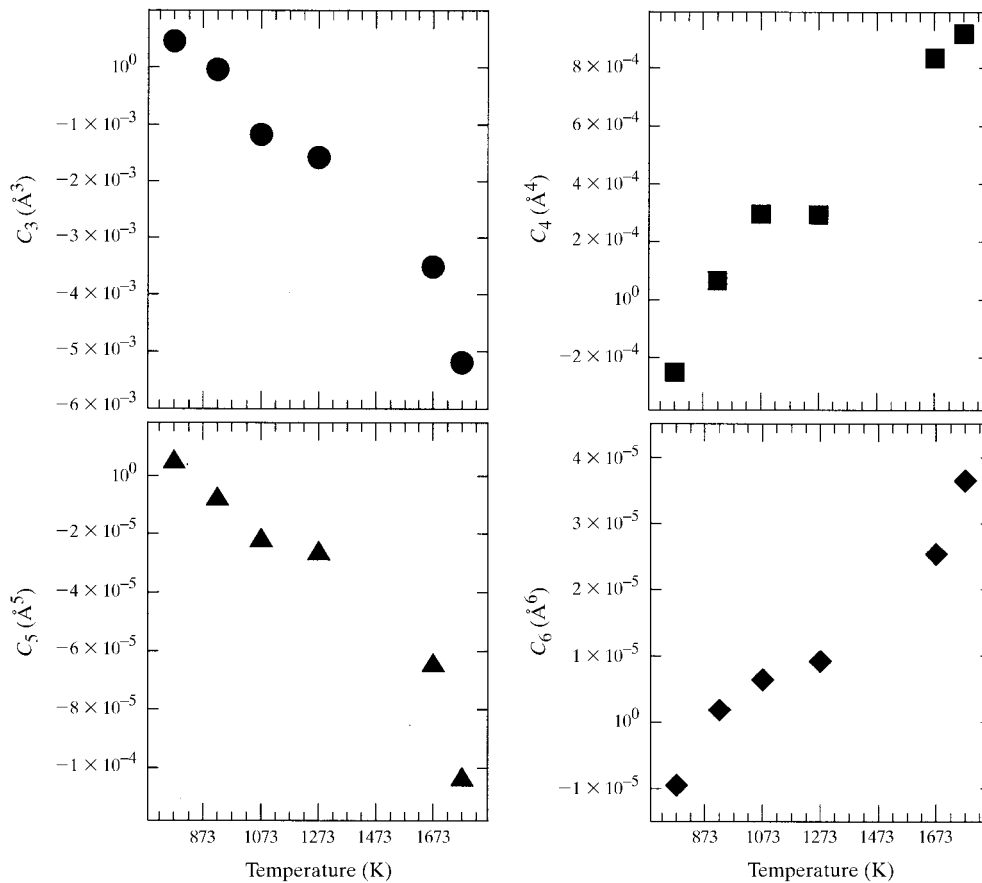


Figure 6

Behaviour of cumulants with temperature.

using the cumulant expansion (anharmonic correction). It appears that, in the harmonic approximation, the coordination number N has a mean value of about 1.52 up to 1523 K, corresponding to a length of Se chains of $n = 4$ through the relation $N = 2 - 2/n$, and decreases below 1 at higher temperatures.

Estimation of the statistical uncertainty has been made according to the method described in §4.2.2. The standard deviations on structural parameters (when ignoring correlations) were obtained from the covariance matrix. They are not dependent on the statistical errors on $k^n\chi(k)$ if the latter are not k -dependent. The total uncertainties on these parameters were obtained by calculating pair correlations between parameters as explained in §4.3.2. Both the standard deviations and the total uncertainties are displayed in Fig. 4. The constant probability ($p = 0.683$) contour for the coordination number and the Debye–Waller factor is shown in Fig. 5, for the point at 923 K. The width and height of the smallest rectangle containing the contour are an estimation of the total uncertainties (errors bars) on each of the two parameters. No estimation of the uncertainties is presently available in the *SEDEM* software when using the cumulant expansion method.

The behaviour of the coordination number with temperature is in disagreement with diffraction experiments, which give a value of $N \simeq 1.7$ in the fluid state, suggesting an average chain length of $n = 7$. Moreover, N should not decrease below unity because $N = 1$ corresponds to dimers and $N = 0$ corresponds to isolated atoms, so that it would correspond to the presence of isolated Se atoms, in contradiction with previous experiments. This, together with the high value of the Debye–Waller factor, stresses the need to go beyond the harmonic approximation.

Below 873 K, no need for an extra cumulant was found, since adding new cumulants did not change the results. At higher temperatures, cumulants were first computed up to C_4 . Their temperature dependence does not appear monotonous, contrary to what is expected according to Dalba & Fornasini (1997). When computed up to C_6 , their

temperature dependence becomes monotonous, as shown in Fig. 6: when the temperature increases, even-order cumulants increase and odd-order cumulants decrease, an indication that the distance distribution deviates more and more from a Gaussian distribution. When computing the profile of the distance distribution, back from the cumulant expansion, the series

$$\exp\left(\sum_{n=0}^{\infty} \{(2ik)^n/n!\}C_n\right)$$

converges gently with cumulants computed up to C_6 . Fig. 3 shows the result of the anharmonic correction (up to C_6) versus the coordination number, in comparison with those in the harmonic approximation. These results have physical coherence: N stays close to 1.75, corresponding to an average length of the chains of $n \simeq 8$ atoms (assuming no interchain bonding), in good agreement with NMR results giving $n \simeq 7$.

These results strongly support a twofold intrachain coordination of the Se atoms, at about 2.3 Å in the metallic phase as well as in the semiconductor phase. Fig. 7 displays examples of the distance distributions built from the cumulant expansion. At high temperature, the distance distribution exhibits a tail on the short-distance side. From that asymmetry one concludes that a significant proportion of chains shorter than the average length $n \simeq 7$ are present. This is in agreement with simulations using a density-functional molecular dynamics method.

Whereas more conclusions may be drawn from that work on selenium in critical conditions, we limit the present example at this point, because our purpose here is not to discuss the case of selenium, but to illustrate the capabilities of the *SEDEM* software.

6. Conclusions

The *SEDEM* software is interested not in revolutionary methods of extracting or modelling EXAFS data, but rather in a good control of each step of the processing, with a special mention of the normalization of the smooth k -dependence of the spectrum with the help of a parameterization of the atomic absorption computed by the *f-Prime* program or by using the Stobbe formula. Parameter determination includes both a ‘distance minimization’ method which uses a gradient algorithm, with a careful evaluation of uncertainties on resulting parameters, and a phase and amplitude decorrelation method, working for a single shell of neighbours, and giving access to non-harmonic distribution of distances with the help of a cumulant expansion.

It is designed for the interpretation of EXAFS data on the basis of the ‘classical’ EXAFS formula, but it is able to use phase and amplitude data computed by programs which compute multiple scattering.

In addition, a provision is made to read data directly on the workstation where they are collected by the *SPEC* program (*Certified Scientific Software*, 1986) at the CRG-IF

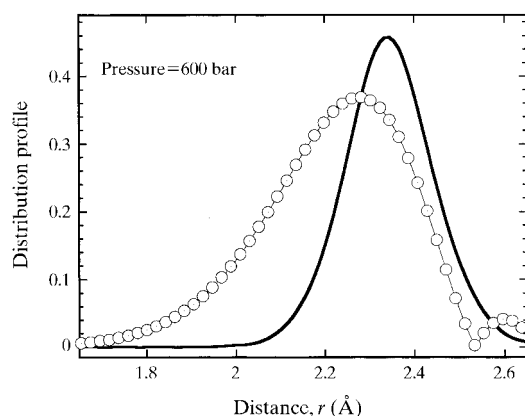


Figure 7

Profiles of the distance distribution function at two different temperatures. At high temperature, a tail towards low r appears. Full line: $T = 773$ K; empty circles: $T = 1673$ K.

absorption instrument at the ESRF, making it a useful tool for rapid data processing during experiments, thus possibly working as an alarm clock.

The author wishes to thank Dominique Bonnin and Alain Michalowicz who gave him access to the sources of their programs and with whom he had very fruitful discussions. He also wishes to thank warmly Jean Louis Hazemann, in charge of the absorption instrument at the CRG-IF, and Yvonne Soldo, who helped a lot in making this program usable and, hopefully, useful.

References

- Certified Scientific Software* (1986). *SPEC*, <http://www.certif.com/>.
- Coy, M., Doyle, C., Besser, J. & Neilands, J. B. (1994). *Biometals*, **7**, 292–298.
- Cromer, D. T. & Liberman, D. (1970). *J. Chem. Phys.* **53**, 1891–1898.
- Dalba, G. & Fornasini, P. (1997). *J. Synchrotron Rad.* **4**, 243–255.
- Dalba, G., Fornasini, P. & Rocca, F. (1993). *Phys. Rev. B*, **47**, 8502–8514.
- Jacquamet, L., Aberdam, D., Adrait, A., Hazemann, J.-L., Latour, J.-M. & Michaud-Soret, I. (1998). *Biochemistry*, **37**, 2564–2571.
- James, R. W. (1950). *The Optical Principles of the Diffraction of X-rays*, ch. IV, p. 157. London: Bell.
- Krill, G. (1986). *Rayonnement Synchrotron dans le Domaine des Rayons X*, Summer School, Aussois, France, September 1986, p. 281.
- Lengeler, B. & Eisenberger, P. (1980). *Phys. Rev. B*, **21**, 4507–4520.
- McKale, A. G., Knapp, G. S. & Chan, S. K. (1988). *J. Am. Chem. Soc.* **110**, 3763–3768.
- McMaster, W. H., Kerr Del Grande, N., Mallett, J. H. & Hubbell, J. H. (1969). Report UCRL-50174, §2. Revision 1. Lawrence Livermore National Laboratory, Livermore, CA, USA.
- Marquardt, D. W. (1963). *J. Soc. Ind. Appl. Math.* **11**, 431–441.
- Michalowicz, A. (1990). Thesis, Université Paris val de Marne, France.
- Press, W. H., Flannery, B. P., Teukolsky, S. A. & Vetterling, W. T. (1986). *Numerical Recipes*, ch. 14. Cambridge University Press.
- Rehr, J. J., Zabinsky, S. I. & Albers, R. C. (1992). *Phys. Rev. Lett.* **69**, 3397–3400.
- Sasaki, S. (1989). KEK Report 88–14. National Laboratory for High Energy Physics, Tsukuba 305, Japan.
- Soldo, Y., Hazemann, J. L., Aberdam, D., Inui, M., Tamura, K., Raoux, D., Pernot, E., Jal, J. F. & Dupuy-Philon, J. (1998). *Phys. Rev. B*, **57**, 258–268.
- Soldo, Y., Hazemann, J. L., Aberdam, D., Pernot, E., Inui, M., Jal, J. F., Tamura, K., Dupuy-Philon, J. & Raoux, D. (1997). *J. Phys. IV*, **7**(C2), 983–985.
- Stern, E. A. (1993). *Phys. Rev. B*, **48**, 9825–9827.
- Stobbe, M. (1930). *Ann. Phys.* **7**, 661.
- Teo, B. K. (1986). *EXAFS, Basic Principles and Data Analysis*, p. 146. Berlin: Springer-Verlag.