MSCD Package Home Page

The MSCD package, which name stands for "Multiple Scattering Calculation of Diffraction", was developed by Yufeng Chen and Michel A Van Hove at Lawrence Berkeley National Laboratory, with several collabarators. This program simulates the elemental and state-specific core-level photoelectron diffraction pattern from a surface, based on multiple scattering theory and the Rehr-Albers separable representation of spherical-wave propagators, and by fitting to experiment obtains atomic structural information.

The program is designed to be very portable, with very simple changes from one computer platform to another. Both parallel and sequential versions are available.

Comprehensive test calculations agree well with more exact (but much slower) calculations performed by other programs. A new version is currently in preparation to take into account photon and electron-spin polarization effects such as circular dichroism.

- The latest version is: Version 1.37

Overview

- Background
- Features
- Version history
- Credits

Multiple Scattering Theory

- Photoemission intensity and the exact representation
- Rehr-Albers separable representation
- Euler angles and rotation matrices
- Composite rotation matrix
- Inelastic scattering
- Correlated vibrational effect
- Inner potential correction
- Instrumental angular averaging

Convergence of the Rehr-Albers Approximation

- Multiple-scattering order
- Rehr-Albers approximation order and initial-state effect
- Pathcut
- Cluster size
- Reliability in structure determination
- Conclusions

Speeding up the calculation

- Iterative summing
- Searching symmetries

Applications

- Calculation and fitting for Ni(001)+c(2x2)-S/S1s
- Calculation and fitting for Fe(001)+c(2x2)-P/P1s
- Calculation and fitting for clean Cu(111)-3s and 3p
- Calculation and fitting for clean Ni(111)-3p

Information on using MSCD

- Source code description
- Executable programs
- Data types and unified data format
- General procedure to make an calculation
- Preparing input data files
- The chi function reliability factor
- Built-in fitting procedure
- Sample data files

Questions and Answers

- Questions and Answers regarding the MSCD package

Obtaining and Installing MSCD

- Download the MSCD source code (C++): MSCD
- Download a program to calculate phase shifts and radial matrix elements (FORTRAN): PSRM (warning: it has no documentation)
- Platform and Requirements
- Installation
- Release Notes
- Known Problems or Bugs

Contact

- Michel A. Van Hove
  Email vanhove@cityu.edu.hk

MSCD Package Overview

Background

The technique of core-level photoelectron diffraction (PD) has been applied to surface structure determination on the atomic scale for more than 20 years. This local diffraction technique probes short-range order around the photoemitter. A broad variety of surfaces have been successfully studied, including metals, semiconductors, oxides, systems exhibiting surface core-level shifts, adsorbed atoms and molecules, epitaxial overlayers, and atoms at buried interfaces. A number of research groups have performed photoelectron diffraction experiments to study surface and interface structures, using both laboratory X-ray and synchrotron radiation sources, and doing the measurements in both scanned-angle and scanned-energy modes. Several reviews of this field have appeared in recent years. A further element which has recently been added to this technique is the application of holographic inversion methods to photoelectron diffraction data so as to more directly yield atomic structures in three dimensions. Thus, the accurate theoretical modeling of photoelectron diffraction is crucial to the development and use of this technique in its various forms, and we here discuss one method for accomplishing this.

In core-level photoemission, a photon illuminates an emitting atom at or near the surface or in the solid and excites an electron from an atomic core-level, ejecting the electron to a detector far away from the surface. The resulting photoelectron wave components can occur both by direct propagation to the far-field detector and via a number of scatterings from atoms in the neighborhood of the emitter. The quantum interference (diffraction) between the different pathways depends sensitively on the relative atomic positions. The intensity associated with this process, when measured as a function of photon energy and/or emission angles, thus includes information about the atomic structure around the emitter. The energy- and angle-resolved photoemission intensity $I(\mathbf{k},\theta,\phi)$ at the detector can be written in general as

$$I(\mathbf{k},\theta,\phi) \propto \left| \phi_0 + \sum_j \phi_{s_j} \right|^2 \qquad (1)$$

where **k** is the final electron wave vector (related to the final kinetic energy), θ and ϕ are the polar and azimuthal angles of photoelectron emission, respectively, ϕ_0 is the wave-component representing travel along a path directly to the detector without being scattered by another atom, and ϕ_{sj} is the wave-component representing travel via paths involving single or multiple scattering by one or more atoms, respectively. The multiple-scattering order is defined as the number of scattering atoms in a path: j = 1 = single, 2 = double, etc. Higher-order scattering processes are less important because of the roughly 1/r falloff of the outgoing distorted spherical wave represented by ϕ_0, damping due to inelastic processes and loss of diffraction modulation due to thermal vibrations (Debye-Waller effects). The explicit sum in Eq. (1) includes all possible scattering paths inside the solid.

To briefly review the history of such theoretical modeling of PD effects, the first qualitative analyses of higher-energy data made use of Kikuchi-band theory. An early quantitative theoretical explanation of these diffraction data was provided by Liebsch in 1974 and then improved by him and others with multiple-scattering effects in 1976; it was based on LEED theory. In the next few years, Pendry, as well as Li, Lubinsky and Tong, put forward similar theories involving the more complex nature of both the initial and final states. These theories require the assumption of full translational symmetry parallel to the surface for the system under investigation.

A single-scattering cluster-based theory based on prior work in extended x-ray absorption fine

structure (EXAFS) and preliminary work on Auger electron diffraction (AED) was first applied to scanned-angle photoelectron diffraction by Kono et al; this was based on the plane-wave approximation. The next major advance in the cluster approach was by Barton and Shirley, who included both spherical-wave corrections and multiple scattering effects. The cluster-based theories are inherently more suitable for photoelectron diffraction modeling in view of the point-source nature of the problem, the spherical outgoing waves involved, the short inelastic attenuation lengths, and the sensitivity to short-range rather than long-range order. In particular, one key advantage of PD as a structure probe is in not requiring long-range translational order, and so being able to model such short-range order structures conveniently without having to resort to some sort of repeated structure with fictitious long-range order is crucial. Cluster-based methods are ideal for this. It has also been found that the maximum cluster size required for accurate simulations is about 100 atoms.

The multiple scattering expansion for spherically-symmetric scatterers is conveniently formulated in terms of diagonal plane-wave scattering t-matrices with elements $t_l = \sin(\delta_l)\exp(i\,\delta_l)$ and matrix elements of the free-particle propagator $G_{L,L'}(\boldsymbol{\rho}) = \langle L,\mathbf{R}|G|L',\mathbf{R'}\rangle$ in an angular momentum $L = (l,m)$ and site $\mathbf{R}$ basis. Here $\rho = k(\mathbf{R}-\mathbf{R'})$ denotes the dimensionless vector between two sites, $k = |\mathbf{k}|$ is the wave number. $G_{LL'}(\boldsymbol{\rho})$ is defined by the following integral involving spherical Bessel functions $j_l(kr)$ and spherical harmonics $Y_L(\hat{\mathbf{k}})$:

$$G_{LL'}(\rho) = -\frac{(4\pi)^2}{2k}\int\frac{d^3\mathbf{k}}{(2\pi)^3}\frac{Y_L^*(\hat{\mathbf{k}})\,Y_{L'}(\hat{\mathbf{k}})\exp(i\mathbf{k}\cdot(\mathbf{R}-\mathbf{R}'))}{\varepsilon - k^2/2 + i0^+}\frac{j_l(kr)j_{l'}(kr')}{j_l(\sqrt{2\varepsilon}\cdot r)j_{l'}(\sqrt{2\varepsilon}\cdot r')} \tag{2}$$

where $\hat{\mathbf{k}}$ is a unit vector along $\mathbf{k}$, r and r' are arbitrary displacements. An (n-1)-atom path (Fig. 1), including the emitter atom at R_0, n-1 scatterers at $\mathbf{R}_1$, $\mathbf{R}_2$, ..., $\mathbf{R}_{n-1}$, and the detector at $\mathbf{R}_n$, is represented by the following total propagator in this *exact multiple-scattering expansion*:

$$G^{(n-1)}_{L_n,L_0}(\mathbf{R}_0,\mathbf{R}_1,\mathbf{R}_2,\ldots,\mathbf{R}_n) = \sum_{(\text{paths})}\sum_{\{L_i\}} G_{L_n,L_{n-1}}(\boldsymbol{\rho}_n)t_{l_{n-1}}(\mathbf{R}_{n-1})G_{L_{n-1},L_{n-2}}(\boldsymbol{\rho}_{n-1})\ldots$$
$$t_{l_2}(\mathbf{R}_2)G_{L_2,L_1}(\boldsymbol{\rho}_2)t_{l_1}(\mathbf{R}_1)G_{L_1,L_0}(\boldsymbol{\rho}_1) \tag{3}$$

where $\boldsymbol{\rho}_i = k(\mathbf{R}_{i+1}-\mathbf{R}_i)$, and L_0 and L_n denote fixed initial and final angular momenta.

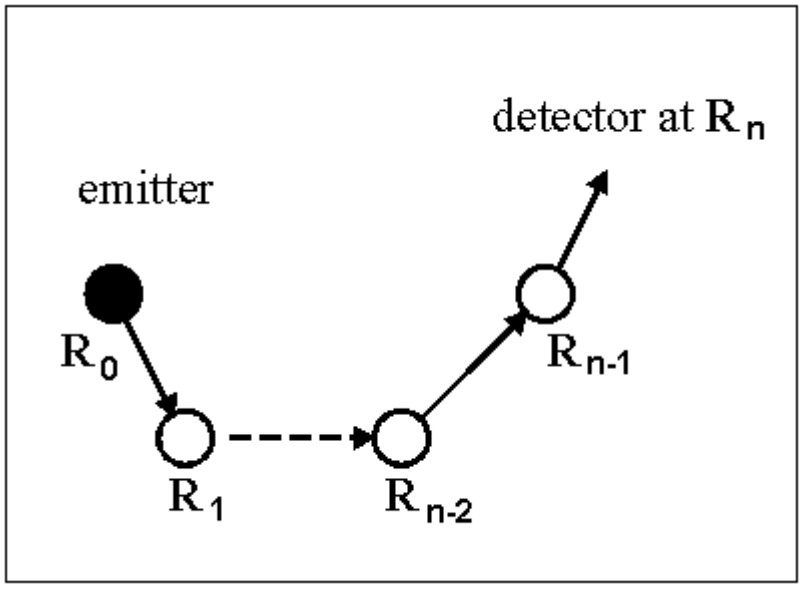


Fig. 1 A photoelectron excited from the core-level of an emitter atom at R_0 propagates to the detector at R_n via n-1 scatterers R_1, R_2, ..., and R_{n-1}. Each straight segment linking consecutive atoms represents a Green's function propagator $G_{LL'}(k(\mathbf{R}_{i+1}-\mathbf{R}_i))$. The emitter can become a scatterer after emission.

Several computer programs have been written to successfully calculate photoelectron diffraction intensities based on this exact cluster formalism, for example, by Chassé and co-workers, and by García de Abajo. However, without special optimization (as we consider below) such exact programs in general require considerable computing times for large clusters, due to the large matrix dimensions involved and the large number of paths that are explicitly summed over. That is, the matrix $G_{LL'}$ has dimensions $(l_{max}+1)^2$ x $(l_{max}+1)^2$, where l_{max} is the largest angular momentum value needed to accurately represent scattering; it can be roughly evaluated using $l_{max} \sim k_{max}R_{mt}$, where R_{mt} is the muffin-tin radius of the scattering potential. For photoelectron diffraction in the energy range 50-1500 eV, this l_{max} lies between 5 and 30, respectively, of course depending on the electron energy. The computing time can be considerably reduced by an iterative summing over the multiple-scattering paths, as proposed by García de Abajo, which leads to proportionality to the number of multiple scatterings, rather than the much steeper behavior as in the traditional summing method. More precisely, the total calculation time in traditional summing is $T \propto T_2^{\,nmax/2}$, while the iterative summing has a total time of $T \propto n_{max}T_2$, where T_2 is the time for a double scattering calculation, and n_{max} is the maximum multiple-scattering order included. A convergent calculation is found to requires n_{max} up to about 6, or even higher in cases with long straight chains for strongly forward scattering atoms. We use these two *exact* formalisms here as references with which to compare calculations performed with the Rehr-Albers expansion.

Several methods have been proposed previously for approximating the exact scattering formalism so as to reduce computation times in cluster calculations.

- Barton and Shirley first derived a representation of the exact formalism with their Taylor-series magnetic-quantum-number expansion (MQNE). This reduces the matrix dimension that needs to be used, since only the first 3 Taylor orders are usually required.
- Fritzsche, Rennert and Chassé provided a reduced angular momentum expansion (RAME) approximation. The incoming spherical waves are approximated by a limited set of spherical

harmonics with low angular momenta (quantum numbers (l,m) = (0,0), (1,-1), (1,0) and (1,1)). Improvement has been achieved by taking into account two more components of higher angular momenta (l,m) = (2,0) and (3,0). A generalized scattering amplitude of each scattering process contains curved wave corrections and a rotation operator which rotates the z-axis between the wave vectors before and after the scattering.

- As another approach to reduce computation times, Rehr and Albers developed a theory, denoted R-A in this paper, that is based on a separable representation of the exact formalism; it was developed primarily to calculate NEXAFS (near-edge x-ray absorption fine structure) and EXAFS (extended x-ray absorption edge fine structure). Curved-wave multiple-scattering contributions can in this approach be calculated with an efficient formalism similar to that based on the plane-wave approximation, but with scattering amplitudes replaced by distance-dependent scattering matrices that act like effective scattering factors. A useful property of this method is that the size of the scattering matrices can be systematically increased as needed for sufficient accuracy, ultimately recovering the exact result. Thereby, the successive orders of scattering can be built up and convergence achieved in a convenient and efficient way. The first application of this theory to photoelectron diffraction was in a computer program developed by Kaduwela, Friedman, and Fadley, and this was based on the second order of the R-A approximation, i.e. scattering matrices of dimension (6x6).
- In a more recent development based on the R-A approach, Wu, Chen and Shirley developed a faster program that benefits from a novel reverse-summation method over paths and an iterative summing technique, but is again limited to second order in R-A. Chen and Van Hove have further developed this method to enable the inclusion up to 4th order in R-A (up to (15x15) matrices). By an efficient path-cut process, this program runs even faster. The resulting computer package, called MSCD, is also very portable, running on a variety of sequential and parallel computers.

By applying object oriented programming within C++, the MSCD code provides a better user interface (particularly important for convenient use by experimentalists), general data formats, and fully functional subsidiary utilities. The simulation program is also designed to be very portable, with very simple changes from one computer type to another. The sequential version of the code has been implemented on supercomputers (under Unix), on Sun workstations (Unix), as well as on PC and Macintosh desktop computers. The parallel version has been implemented on a massively-parallel supercomputer (Cray T3E) and on a cluster of multiprocessor systems (COMPS, a type of network of workstations) under MPI.

Comprehensive test calculations agree well with more exact (but much slower) calculations performed by other programs. A new version is currently being generalized to take into account photon and electron-spin polarization effects such as circular dichroism (the latter is obtained by contrasting measurements performed with right- vs. left-circular polarized light), magnetism (including magnetic circular dichroism), and relativistic effects such as spin flip.

In the next sections, we will briefly introduce both the exact and R-A formalisms, discuss the methods that have been used to increase the calculational efficiency of R-A, present a number of results calculated with the MSCD program based on R-A, and compare these results with exact cluster calculations to quantitatively evaluate the performance and limitations of the R-A approximation as carried out to different orders. We will discuss a few more necessary details about the exact cluster-based formalism, and briefly review the R-A method and its properties. The methodology and convergence of the implementation of R-A used in the MCSD will be presented, as well as the overall reliability of the MSCD approach for structural determinations.

MSCD Features

Table of some photoelectron diffraction simulation programs (as of 8/1/98)

Code	mscd	scat	txpd	newchp	spdl	javier
Version	1.37/98	3.65/96	95	91	97	97
Language	C++	Fortran	Fortran	Fortran	Fortran	C
Method	R-A approx	R-A approx	R-A approx	JB approx	Exact	Exact
Platform	PC/Mac/Sun/ Cray/Comps	PC/Mac/Sun/ Cray	Sun/Cray	Mac	Sun/Cray	Sun
Parallel	Yes	No	No	No	No	No
Ini. State	all	all	all	S only	all	all
Scanning	energy/angular	energy/angular	angular	energy	energy/angular	energy
Rotation	analyzer/sample	analyzer/sample	sample	N/A	analyzer	N/A
Hologram	Yes	Yes	Yes	No	No	No
Fitting	Yes	Yes	No	No	No	No
Vibration	correlated	correlated	correlated	uncorrel.	uncorrel.	uncorrel.
Polarization	linear	linear	linear/circular	linear	linear/circular	linear
MLD/MCD	No	No	Yes	No	Yes	No

MSCD Version History

The programming of the MSCD package began on December 9, 1996, using the ANSI C++ object oriented programming (OOP) technique and message passing interfacing (MPI) parallel communication.

- Version 1.37, June 10, 1998
  Make the source code compatible to Macintosh CodeWarrier Compiler.
  Fixed a bug in net search mode.
- Version 1.36, June 1, 1998
  Impoved emission angle search feature.
- Version 1.35, April 21, 1998
  Number of emitters in different logical layers is considered. Total number of data points can be up to 30000.
- Version 1.34, February 22, 1998
  Fixed a bug in calculation of large cluster.
- Version 1.32, January 16, 1997
  Fixed a bug in calculation of reliability factors for two hologram data files. Calculate Fadley and Van Hove group's reliability factors in program caldif
- Version 1.31, August 31, 1997
  Expand the Rehr-Albers approximation order from 3 (10x10 matrix) to 4 (15x15 matrix). To choose 4th order of Rehr-Albers approximation, set raorder=4 in the input file.
- Version 1.30, August 30, 1997
  Expand the Rehr-Albers approximation order from 2 (6x6 matrix) to 3 (10x10 matrix). To choose 3rd order of R-A approximation, set raorder=3 in the input file.
- Version 1.24, June 30, 1997
  1) Add usermac.cpp file. The program works on Macintosh computer. 2) Add two functions in userutil.cpp, skipendline and getendline, making the program package support PC, Unix and Macintosh ascii text format data file. 3) Add usercomp.cpp file. The program works on COMPS network of workstation in parallel mode. 4) Add cpu time analysis function for

each processor in parallel mode. The processor times are divided into four categories: computation, sending, receiving and idle. To activate this feature, set display mode to 10. The total processor time and its distribution are shown in output file mscdlist.txt.

- Version 1.23, June 12, 1997
  The maxmum number lmax of angular momentum is expanded again to 60. To make it possible, the makecurve function in rotamat.cpp has been changed, using double precission floating point numbers.
- Version 1.22, June 10, 1997
  The maximum energy and number of angular momentum are expanded. The maximum kmax set to 25.0, corresponding to energy 2.38 keV. The maximum lmax set to 30.
- Version 1.21, June 3, 1997
  Removed a bug in fitting mode and a possible bug in test-running mode.
- Version 1.20, May 28, 1997
  First parallel version, working on Cray T3E in MPP parallel mode.
- Version 1.10, April 1, 1997
  Introduce the rotation transfer method to save computation time for circular polarized light and magnetic circular dichroism.
- Version 1.00, March 1, 1997
  First version, working on PC, Sun workstation, Cray J90 supercomputer, and Cray T3E supercomputer in sequential mode, covering all the features of the SCAT Version 3.65 package.

Credits and Acknowledgements

We thank Dr. H. Wu, Dr. Z. Hussain and D.A. Shirley for their collaboration in developing the SCAT package. The MSCD package uses the same method and similar data format, and this documentation takes some contents from the SCAT package user guide.

We thank our colleagues in D.A. Shirley's group at Lawrence Berkeley National Laboratory, E. Moler, S. Kellar, W.A. Huff, X. Zhou, and B. Peterson, for their detailed discussions and many test calculations. Special thanks also go to C.S. Fadley, J.J. Rehr and A. Kaduwela for many helpful discussions regarding photoelectron diffraction theory and its computation. R.X. Ynzunza, F.J. García de Abajo and A. Chassé ran many time consuming test calculations and comparisons with other codes, which helped us to debug and improve the software

Multiple Scattering Theory

Photoemission intensity and the exact representation

The term *exact* in this document means that no R-A approximation is made, i.e., that, even if a R-A expansion is used, it is carried to all orders, since it will in principle converge to the *exact* result. We note in this context that Brouder and Sébilleau have shown that the R-A representation carried to all orders turns out to be an accurate, stable and efficient way to calculate the exact propagator.

For simplicity in the formulas of photoemission intensity, we do not include effects due to inelastic scattering or vibrational motion, even though it is clear that these effects are essential for a quantitative description of experimental data. Both of these effects tend to damp out scattering from atoms further from the emitter, thus causing multiple-scattering paths to be effectively reduced in length. Both of these effects are, however, included in our MSCD programs and in some of the calculations presented in this document. Using the exact propagator,

$$G^{(n-1)}_{L_n,L_0}(\mathbf{R}_0,\mathbf{R}_1,\mathbf{R}_2,\ldots,\mathbf{R}_n) = \sum_{(\text{paths})}\sum_{\{L_i\}} G_{L_n,L_{n-1}}(\boldsymbol{\rho}_n)t_{l_{n-1}}(\mathbf{R}_{n-1})G_{L_{n-1},L_{n-2}}(\boldsymbol{\rho}_{n-1})\ldots$$
$$t_{l_2}(\mathbf{R}_2)G_{L_2,L_1}(\boldsymbol{\rho}_2)t_{l_1}(\mathbf{R}_1)G_{L_1,L_0}(\boldsymbol{\rho}_1) \qquad (3)$$

the multiple scattering photoemission intensity can be expressed as:

$$I^{(n_{max})}_{n_i l_i}(k,\theta,\phi) \propto \sum_{\text{emitter}}\sum_{m_i}\left|\sum_{l_f=l_i\pm1} m_{l_f,c}\exp(i\delta_{l_f,c})\times\right.$$
$$\left.\left[G^{(0)}_{00,l_f m_i}(\mathbf{R}_0,\mathbf{R}_d) + G^{(1)}_{00,l_f m_i}(\mathbf{R}_0,\mathbf{R}_1,\mathbf{R}_d) + \sum_{n=3}^{n_{max}} G^{(n-1)}_{00,l_f m_i}(\mathbf{R}_0,\mathbf{R}_1,\mathbf{R}_2,\cdots,\mathbf{R}_{n-1},\mathbf{R}_d)\right]\right|^2 \qquad (4)$$

where $I^{(n_{max})}_{n_i l_i}(k,\theta,\phi)$ is the photoemission intensity from electronic subshell (n_i,l_i), as detected with wave vector k in the (θ, ϕ) direction; (n_i,l_i,m_i) are the quantum numbers of the initial core orbital (n_i = 1, 2, 3, 4, ... = K,L,M,N,... shells, respectively, l_i = 0, 1, 2, 3, ... = s, p, d, f, ... subshells, and m_i is the magnetic quantum number); $L_f = (l_f,m_f)$ are angular momentum quantum numbers of the final state. The dipole selection rules imply that $l_f - l_i = \pm1$, $m_f - m_i = 0$, where we for simplicity restrict ourselves to linear polarized incident light, although this can be generalized in a straightforward way to other polarizations, such as circular or elliptical. $G^{(n-1)}_{00,l_f m_i}(\mathbf{R}_0,\mathbf{R}_1,\mathbf{R}_2,\cdots,\mathbf{R}_{n-1},\mathbf{R}_d)$ is the exact nth-order multiple scattering Green's function for a scattering path from the emitter at $\mathbf{R}_0 \equiv \mathbf{R}_{emitter}$ via scatterers at $\mathbf{R}_1$, $\mathbf{R}_2$, ... , $\mathbf{R}_{n-1}$ to the detector at $\mathbf{R}_n \equiv \mathbf{R}_{detector} \equiv \mathbf{R}_d$. The quantities $m_{lf,c}$ and $\delta_{lf,c}$ are the amplitude and phase of the dipole matrix element into a given final state, and are related to the short-range central potential of the ionized atom; that is, the long-range Coulomb field is neglected due to the assumed screening of the core hole near a solid surface. The quantities $m_{lf,c}$ and $\delta_{lf,c}$ are calculated from $\langle\Psi_{Ekin,lf}|\ \varepsilon\cdot\mathbf{r}|\ \phi_{nili}\rangle$, with $\Psi_{Ekin,lf}$ is the final continuum state of the photoelectron at a kinetic energy E_{kin} which propagates in direction $\mathbf{r}$, ϕ_{nili} is the initial core orbital $L_i = (l_i, m_i)$ from which the photoelectron is emitted, and ε is the radiation polarization vector. The summations run over all emitters, all final states $L_f = (l_f, m_f)$ and over all combinations of order n, the number of scatterers (number of atoms = n+1) in a given scattering path from single scattering (n = 1) to the highest order considered $n = n_{max}$ (typically we select n_{max} = 8 or higher, corresponding to 7 or more scattering events).

Now we choose the **z** direction to be parallel to the ε vector, to simplify the matrix element evaluation. The expression for the matrix element then becomes:

$$m_{l_f,c} = (-i)^{l_f} \; < R_{E_{kin},l_f}(r) \,|\, r \,|\, R_{n_i,l_i}(r) > \quad < Y_{l_f,m_i}(\theta,\phi) \,|\, Y_{10}(\theta,\phi) \,|\, Y_{l_i,m_i}(\theta,\phi) > \tag{5}$$

where $R_{Ekin,lf}(r)$ is the radial part of the continuum orbital at l_f, $R_{nili}(r)$ is the radial part of the initial core orbital with quantum numbers n_i and l_i, and $Y_{lm}(\theta,\phi)$ are the relevant spherical harmonics.

Rehr-Albers separable representation

In the convergent separable representation of the propagator derived by Rehr and Albers, the exact propagator $G_{LL'}$ is re-written as

$$G_{LL'}(\boldsymbol{\rho}) = \frac{\exp(i|\boldsymbol{\rho}|)}{|\boldsymbol{\rho}|} \sum_{\lambda} \tilde{\Gamma}^{L}_{\lambda}(\boldsymbol{\rho}) \Gamma^{L'}_{\lambda}(\boldsymbol{\rho}) \tag{6}$$

where a new combination index $\lambda = (\mu,\nu)$ is introduced, such that, for an exact representation, $\mu = -l_{max}$ to l_{max}, and $\nu = 0$ to $|\mu|$. However, this expansion converges relatively quickly, and can usually be truncated without significant loss of accuracy, as we shall show later. The quantities $\Gamma^{L}_{\lambda}(\boldsymbol{\rho})$ and $\tilde{\Gamma}^{L}_{\lambda}(\boldsymbol{\rho})$ have the following forms:

$$\Gamma^{L}_{\lambda}(\boldsymbol{\rho}) = (-1)^{\mu} N_{l\mu} \frac{C_1^{(|\mu|+\nu)}(z)}{(|\mu|+\nu)!} z^{|\mu|+\nu} R^{1}_{\mu m}(\Omega_\rho) \tag{7}$$

$$\tilde{\Gamma}^{L}_{\lambda}(\boldsymbol{\rho}) = R^{1}_{m\mu}(\Omega_\rho^{-1}) \frac{(2l+1)}{N_{l\mu}} \frac{C_1^{(\nu)}(z)}{\nu!} z^{\nu} \tag{8}$$

with

$$N_{l\mu} = [(2l+1)(1-|\mu|)!/(1+|\mu|)!]^{1/2} \tag{9}$$

$$C_1^{\nu}(z) = \frac{d^{\nu}}{dz^{\nu}} C_1(z) \tag{10}$$

Here $z = 1/(i|\boldsymbol{\rho}|)$, $C_l(z)$ is the degree-l polynomial factor of the spherical Hankel function, $R^{1}_{\mu m}(\Omega_\rho)$ is a matrix which rotates the bond direction **ρ** onto the **z** axis, and Ω_ρ represents the Euler angles for this rotation. The matrix $R^{1}_{\mu m}(\Omega_\rho)$ transforms the spherical harmonics as

$$Y_{lm}(\hat{\mathbf{k}}) = \sum_{m'} Y_{lm'}(\hat{\mathbf{k}}') R^{1}_{m'm}(\Omega_\rho) \tag{11}$$

Substituting Eq. (6) into Eq. (3), one thus obtains the exact equivalent form

$$G^{(n-1)}_{00,L_f}(\mathbf{R}_1,\mathbf{R}_2,\cdots,\mathbf{R}_n) = \sum_{(paths)} \sum_{\{\lambda_i\}} \tilde{\Gamma}^{00}_{\lambda_n}(\boldsymbol{\rho}_n) F_{\lambda_n\lambda_{n-1}}(\boldsymbol{\rho}_n,\boldsymbol{\rho}_{n-1}) \cdots \times$$
$$F_{\lambda_3\lambda_2}(\boldsymbol{\rho}_3,\boldsymbol{\rho}_2) F_{\lambda_2\lambda_1}(\boldsymbol{\rho}_2,\boldsymbol{\rho}_1) \Gamma^{L_f}_{\lambda_1}(\boldsymbol{\rho}_1) \tag{12}$$

This is the Rehr-Albers separable representation formula for curved-wave multiple-scattering, which is a direct analog of the plane-wave approximation or the point-scattering approximation.

In Eq. (12), the *scattering-amplitude matrices* $F_{\lambda j,\lambda\, j-1}(\boldsymbol{\rho}_j, \boldsymbol{\rho}_{j-1})$ at each site are defined in the partial-wave expansion as

$$F_{\lambda\lambda'}(\boldsymbol{\rho},\boldsymbol{\rho}') = \frac{\exp(i|\boldsymbol{\rho}'|)}{|\boldsymbol{\rho}'|}\sum_L t_l \Gamma_\lambda^L(\boldsymbol{\rho})\,\tilde{\Gamma}_{\lambda'}^{L'}(\boldsymbol{\rho}') \tag{13}$$

where $\boldsymbol{\rho}'$ and $\boldsymbol{\rho}$ are the interatomic vectors leading from and to the site, as illustrated in Fig. 2. The sum on L runs over both l and m quantum numbers.

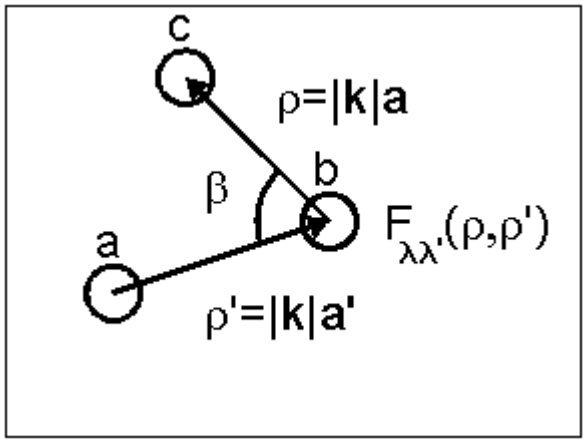


Fig. 2. A scattering event leading from atom a to atom c via atom b; $\boldsymbol{\rho}' = k\mathbf{a}'$ and $\boldsymbol{\rho} = k\mathbf{a}$ are dimensionless interatomic vectors leading from and to the site in question, with $\mathbf{a}'$ and $\mathbf{a}$ the corresponding vectors and k the wave number ($k = |\mathbf{k}|$, where $\mathbf{k}$ is the wave vector), while β is the angle between the interatomic vectors $\boldsymbol{\rho}'$ and $\boldsymbol{\rho}$. The quantity $F_{\lambda\lambda'}(\boldsymbol{\rho},\boldsymbol{\rho}')$ is the effective scattering amplitude via the Rehr-Albers approximation.

The R-A representation, Eq. (13), is thus an exact formula if we take all the possible $\lambda = (\mu,\nu)$ values into account. But in practice, noting the asymptotic form $F_{\lambda\lambda'} \propto (\rho)^{-(2\nu+\mu)} \cdot (\rho')^{-(2\,\nu'+\mu')}$ for large ρ' and ρ, we can safely truncate $F_{\lambda\lambda'}$ at different approximation orders. Table 1 lists, as a function of the R-A approximation order, the dimensions of the scattering-amplitude matrices, and their possible (μ,ν) values. For most real cases that we have encountered, it was found that the second order ((6x6) matrices) is adequate to simulate experimental curves, and this will be further investigated in later sections.

The advantage of the R-A representation is that the approximation leads to smaller matrix sizes, resulting in much reduced computation times. In the exact formalism, Eq. (3), the propagator matrix $G_{LL'}$ has the dimensions $(l_{max}+1)^2$ by $(l_{max}+1)^2$, where l_{max} can be estimated as described previously; for a typical muffin-tin radius R_{mt} of 1.5 Å this yields matrix sizes of (36x36) to (441x441) from low to high energies. By contrast, in most cases, the R-A representation requires matrix sizes of only (6x6), although we discuss below some cases where going up to (15x15) might be required for ultimate quantitative accuracy.

R-A order	F matrix dimension	(μ,ν)
0	1 x 1	(0,0)
1	3 x 3	(0,0), (±1,0)
2	6 x 6	(0,0), (±1,0), (0,1), (±2,0)
3	10 x 10	(0,0), (±1,0), (0,1), (±2,0), (±1,1), (±3,0)
4	15 x 15	(0,0), (±1,0), (0,1), (±2,0), (±1,1), (±3,0), (0,2), (±2,1), (±4,0)

Table 1. Rehr-Albers approximation orders, dimensions of the corresponding scattering-amplitude matrices, and allowed values of (μ, ν).

Euler angles and rotation matrices

In equation (11), we defined the rotation Ω_ρ as a sequence of rotations with three Euler angles α, β, and γ. There are, however, several conventions in existence for choosing the so-called Euler angles. We here adopt the convention used by A. Messiah, which differs slightly from the one generally adopted in the theory of the gyroscope.

The general displacement of a rigid body due to a rotation about a fixed point may be obtained by performing three Euler rotations about two of three mutually perpendicular axes fixed in the body. We assume a right-handed frame of axes, and define a positive rotation about a given axis to be one which would carry a right-handed screw in the positive direction along that axis. Thus a rotation about the z-axis which carried the x-axis into the original position of the y-axis would be considered to be positive. The rotations are to be performed successively in the following order (see Fig. 4):

1. a rotation by α ($0 =< \alpha <= 2\pi$) about the z-axis, bringing the frame of axes and the body together from the initial position S into the position S'.
2. a rotation by β ($0 =< \beta <= \pi$) about the y-axis of the frame S' with its body, resulting in a new position S".
3. a rotation by γ ($-\pi =< \gamma <= \pi$) about the z-axis of the frame S" with its body. The position of this axis depends on the previous rotations α and β. The final position of the frame is symbolized by S'".

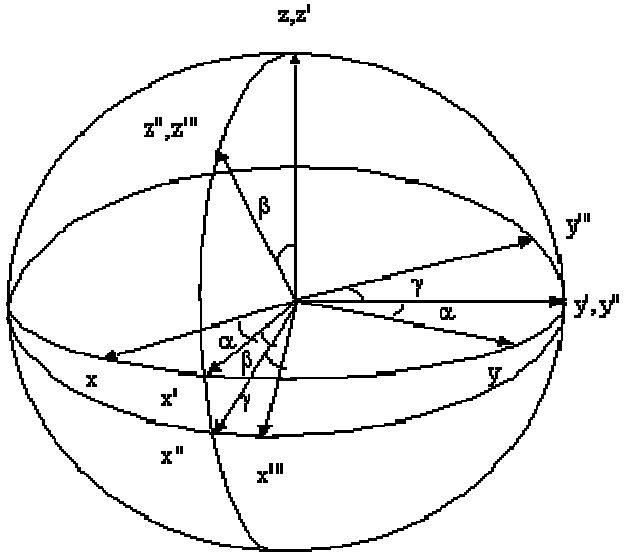


Fig. 3. Definition of Euler angles.

Using this definition, we have the rotation matrices

$$R^{j}_{mm'}(\alpha,\beta,\gamma) = \exp(-im\alpha)\, r^{j}_{mm'}(\beta)\exp(-im'\gamma) \tag{14}$$

Here $r^{j}_{mm'}(\beta)$ is an real and unitary matrix, which has the Wigner formula

$$r^{j}_{mm'}(\beta) = \sum_{t=0}^{\min(j\pm m, j\pm m')} (-1)^{t} \frac{\sqrt{(j+m)!(j-m)!(j+m')!(j-m')!}}{(j+m-t)!(j-m'-t)!t!(t-m+m')!} \times \tag{15}$$

$$(\cos(\beta/2))^{2j+m-m'-2t}(\sin(\beta/2))^{2t-m+m'} \qquad (for\ m' \geq |m|)$$

Equation (15) is only valid for (m' >= |m|). For other combinations of m and m', we can use the following symmetry properties:

$$r^{j}_{mm'}(\beta) = r^{j}_{m'm}(-\beta) = (-1)^{m-m'} r^{j}_{m'm}(\beta) = (-1)^{m-m'} r^{j}_{-m-m'}(\beta) \tag{16}$$

For the case l=1, we have

$$r^{1}(\beta) = \begin{pmatrix} \frac{(1+\cos\beta)}{2} & -\frac{\sin\beta}{\sqrt{2}} & \frac{(1-\cos\beta)}{2} \\ \frac{\sin\beta}{\sqrt{2}} & \cos\beta & -\frac{\sin\beta}{\sqrt{2}} \\ \frac{(1-\cos\beta)}{2} & \frac{\sin\beta}{\sqrt{2}} & \frac{(1+\cos\beta)}{2} \end{pmatrix} \tag{17}$$

In this expression, the successive lines correspond to m=1,0,-1; the columns are arranged in the same order from left to right.

Composite rotation matrix

Substituting equations (7) and (8) into equation (13), we can re-express the scattering-amplitude matrix $F_{\lambda\lambda'}(\boldsymbol{\rho},\boldsymbol{\rho}')$ as

$$F_{\lambda\lambda'}(\rho,\rho') = \frac{\exp(i\rho')}{\rho'}\sum_l t_l \gamma^l_{\mu\nu}(\rho) R^l_{\mu\mu'}(\Omega_{\rho\rho'}) \tilde{\gamma}^l_{\mu'\nu'}(\rho') \tag{18}$$

Here, we introduced the composite rotation matrix

$$R^l_{\mu\mu'}(\Omega_{\rho\rho'}) = \sum_m R^l_{\mu m}(\Omega_\rho) R^l_{m\mu'}(\Omega^{-1}_{\rho'}) \tag{19}$$

and

$$\tilde{\gamma}^l_{\mu\nu}(\rho) = \frac{(2l+1)}{N_{l\mu}} \frac{C^{(\nu)}_l(z)}{\nu!} z^\nu \qquad \gamma^l_{\mu\nu}(\rho) = (-1)^\mu N_{l\mu} \frac{C^{(\mu+\nu)}_l(z)}{(\mu+\nu)!} z^{\mu+\nu} \tag{20}$$

where $R^1_{\mu m}(\Omega_\rho)$ is a rotation taking bond vector $\boldsymbol{\rho}$ into $\mathbf{z}$, and $R^l_{m\mu'}(\Omega^{-1}_{\rho'})$ is a rotation taking $\mathbf{z}$ into bond vector $\boldsymbol{\rho}'$. So we define the composite rotation matrix $R^l_{\mu\mu'}(\Omega_{\rho\rho'})$ which takes $\boldsymbol{\rho}$ into $\mathbf{z}$ and then $\mathbf{z}$ into $\boldsymbol{\rho}'$. By choosing the photon polarization direction as the $\mathbf{z}$ direction, we have $\Omega_\rho = (0,\theta,\pi-\phi)$, $\Omega_{\rho'}^{-1} = (\phi'-\pi, -\theta', 0)$. Let $\Omega_{\rho\rho'} \equiv (\alpha, \beta, \gamma)$, then

$$R^l_{\mu\mu'}(\alpha,\beta,\gamma) = \sum_m R^l_{\mu m}(0,\theta,\pi-\phi) R^l_{m\mu'}(\phi'-\pi,-\theta',0) \tag{21}$$

Now we can derive the composite rotation Euler angles (α, β, γ) using this equation for the case l=1. Substituting equations (14) and (17) into equation (21) for l=1:

$$\begin{bmatrix} \frac{(1+\cos\beta)}{2}\exp(-i(\alpha+\gamma)) & -\frac{\sin\beta}{\sqrt{2}}\exp(-i\alpha) & \frac{(1-\cos\beta)}{2}\exp(-i(\alpha-\gamma)) \\ \frac{\sin\beta}{\sqrt{2}}\exp(-i\gamma) & \cos\beta & -\frac{\sin\beta}{\sqrt{2}}\exp(i\gamma) \\ \frac{(1-\cos\beta)}{2}\exp(i(\alpha-\gamma)) & \frac{\sin\beta}{\sqrt{2}}\exp(i\alpha) & \frac{(1+\cos\beta)}{2}\exp(i(\alpha+\gamma)) \end{bmatrix} =$$

$$\begin{bmatrix} \frac{(1+\cos\theta)}{2}\exp(-i(\pi-\phi)) & -\frac{\sin\theta}{\sqrt{2}} & \frac{(1-\cos\theta)}{2}\exp(i(\pi-\phi)) \\ \frac{\sin\theta}{\sqrt{2}}\exp(-i(\pi-\phi)) & \cos\theta & -\frac{\sin\theta}{\sqrt{2}}\exp(i(\pi-\phi)) \\ \frac{(1-\cos\theta)}{2}\exp(-i(\pi-\phi)) & \frac{\sin\theta}{\sqrt{2}} & \frac{(1+\cos\theta)}{2}\exp(i(\pi-\phi)) \end{bmatrix} x \qquad (22)$$

$$\begin{bmatrix} \frac{(1+\cos\theta')}{2}\exp(-i(\phi'-\pi)) & \frac{\sin\theta'}{\sqrt{2}}\exp(-i(\phi'-\pi)) & \frac{(1-\cos\theta')}{2}\exp(-i(\phi'-\pi)) \\ -\frac{\sin\theta'}{\sqrt{2}} & \cos\theta' & \frac{\sin\theta'}{\sqrt{2}} \\ \frac{(1-\cos\theta')}{2}\exp(i(\phi'-\pi)) & -\frac{\sin\theta'}{\sqrt{2}}\exp(i(\phi'-\pi)) & \frac{(1+\cos\theta')}{2}\exp(i(\phi'-\pi)) \end{bmatrix}$$

From $R_{22}(\Omega_{\rho\rho'}) = \Sigma\, R_{2m}(0,\theta,\theta-\phi)\, R_{m2}(\phi'-\theta,-\theta',0)$ we obtain

$$\cos\beta = \cos\theta\cos\theta' + \sin\theta\sin\theta'\cos(\phi'-\phi) \qquad (23)$$

Here we can get the conditions for $\beta=0$ or θ, as follows:

$\beta=0$ only when (i) $\theta'=\theta$, $\phi'=\phi$ or (ii) $\theta'=\theta=0$ or (iii) $\theta'=\theta=\theta$, and

$\beta=\theta$ only when (i) $\theta'+\theta=\theta$, $\phi'-\phi=\theta$ or (ii) $\theta'=\theta$, $\theta=0$ or (iii) $\theta'=0$, $\theta=\theta$

Evaluating the matrix elements $R_{22}(\Omega_{\rho\rho'})$, $R_{-10}(\Omega_{\rho\rho'})$, $R_{0-1}(\Omega_{\rho\rho'})$, $R_{-1-1}(\Omega_{\rho\rho'})$, and $R_{-11}(\Omega_{\rho\rho'})$, assuming $\gamma=0$ and using the above conditions when $\beta=0$ or θ, we finally obtain the following expression for the Euler angles ($\alpha\beta\gamma$) of the composite rotation,

$$\begin{aligned} &\beta = \cos^{-1}(\cos\theta\cos\theta' + \sin\theta\sin\theta'\cos(\phi'-\phi)) \\ &\alpha = \arg[\sin\theta\cos\theta' - \cos\theta\sin\theta'\cos(\phi'-\phi) - i\sin\theta'\sin(\phi'-\phi)] \quad (for\ 0<\beta<\pi) \\ &\gamma = \arg[\sin\theta\cos\theta'\cos(\phi'-\phi) - \cos\theta\sin\theta' + i\sin\theta\sin(\phi'-\phi)] \quad (for\ 0<\beta<\pi) \\ &\alpha = (\phi'-\phi)\,sign(1.0,\cos\theta), \qquad \gamma = 0 \qquad (for\ \beta=0\ or\ \beta=\pi) \end{aligned} \qquad (24)$$

with

$$sign(1.0,\cos\theta) = \begin{cases} +1.0 & for\ \cos\theta \geq 0 \\ -1.0 & for\ \cos\theta < 0 \end{cases} \qquad (25)$$

Inelastic scattering

A fully rigorous method for including inelastic attenuation is so far not available, and thus we use the common phenomenological approach of an exponential decay of the amplitude of each component of the photoelectron wave with the distance traveled in the solid before escaping through the surface, called electron inelastic mean free path (IMFP). If the distance traveled along a given path is *a* and IMFP is λ, then the exponential decay factor for the amplitude of this path is exp(-*a*/(2λ)). Considering the inelastic scattering and vibrational effects, we only need to rewrite eqn. (18) as follows:

$$F_{\lambda\lambda'}(\alpha,\alpha') = \exp(-i\mu\alpha - i\mu'\gamma') \times \exp\left(-\frac{a'}{2\lambda} - k^2(1-\cos\beta)\sigma_c^2\right) \times \frac{\exp(i\alpha')}{\alpha'} \sum_l t_l \gamma^l_{\mu\nu}(\alpha) d^l_{\mu\mu'}(\beta) \tilde{\gamma}^l_{\mu'\nu'}(\alpha') \tag{26}$$

where a' is the internuclear distance of the bond vector leading from the site in a single scattering process, see Figure 2. σ_c^2 is the thermal mean square relative displacement, which will be discussed later.

The inelastic mean free path can be obtained from theory and certain types of experiments. Tanuma, Powell and Penn have found an empirical formula, called TPP-2 formula, to calculate IMFP for 50-2000 eV electrons, based on the assumption that the Born approximation is valid and on the neglect of vertex corrections, self-consistency, exchange and correlation. The TPP-2 formula is

$$\lambda = E / \{E_p^2[\beta\log(\gamma E) - (C/E) + (D/E^2)]\} \tag{27}$$

where λ is the IMFP (in Å), E is the electron energy (in eV), E_p=28.821(N_v ρ/M)$^{1/2}$ is the free-electron plasmon energy (in eV), ρ is the density of the bulk (in g.cm^{-3}), N_v is the number of valence electrons per atom (for elements) or molecule (for compounds) and M is the atomic or molecular weight. The terms β, γ, C and D are parameters given by

$$\begin{aligned} \beta &= -0.0216 + 0.994/(E_p^2 + E_g^2)^{1/2} + 7.39\times10^{-4}\rho \\ \gamma &= 0.191\rho \\ C &= 1.97 - 0.91N_v\rho/M = 1.97 - 1.096\times10^{-3}E_p^2 \\ D &= 53.4 - 20.8N_v\rho/M = 53.4 - 0.025E_p^2 \end{aligned} \tag{28}$$

and E_g is the bandgap energy (in eV) for non-conductors, and equals zero for conductors. The relationship of electron energy E and wave number k is

$$k = 0.512331\sqrt{E} \qquad and \qquad E = 3.80977\,k^2 \tag{29}$$

Although there is no known physical basis, there is another relatively simple and convenient means for expressing IMFP dependence on electron energy, which was proposed by Wagner, Davis and Riggs.

$$\lambda = kE^m \tag{30}$$

where k and m are material-dependent parameters. They found that m ranged from 0.54 to 0.81. Generally, similar results have been reported by others. Tables 2 and 3 list the empirical parameters k and m for 27 elements and 15 inorganic compounds over the electron energy range 500-2000 eV. It appears from Tables 2 and 3 that m=0.75±0.03 (one standard deviation) is a reasonable approximation for this group of materials over 50-2000 eV.

Element	k	m	Element	k	m
C	0.129	0.775	Ru	0.0843	0.752
Mg	0.112	0.789	Rh	0.0812	0.747
Al	0.0920	0.777	Pd	0.104	0.748
Si	0.116	0.775	Ag	0.0924	0.730
Ti	0.104	0.783	Hf	0.156	0.719
V	0.0998	0.775	Ta	0.104	0.720
Cr	0.0858	0.763	W	0.0958	0.716
Fe	0.0897	0.753	Re	0.0804	0.713
Ni	0.0942	0.734	Os	0.0990	0.706
Cu	0.107	0.729	Ir	0.104	0.708
Y	0.117	0.768	Pt	0.0956	0.714
Zr	0.104	0.768	Au	0.0951	0.713
Nb	0.132	0.745	Bi	0.118	0.746
Mo	0.0941	0.748			

Table 2. Values of the parameters k and m in the fits of eqn (30) to IMFPs calculated from experimental optical data for 27 elements over the electron energy range 500-2000 eV

Compound	k	m	Compound	k	m
Al_2O_3	0.122	0.750	NaCl	0.192	0.760
GaAs	0.235	0.725	PbS	0.121	0.765
GaP	0.144	0.755	PbTe	0.114	0.771
InAs	0.192	0.736	SiC	0.104	0.764
InP	0.0977	0.761	Si_3N_4	0.136	0.751
InSb	0.196	0.749	SiO_2	0.150	0.764
KCl	0.169	0.769	ZnS	0.145	0.752
LiF	0.127	0.764			

Table 3. Values of the parameters k and m in the fits of eqn (30) to IMFPs calculated from experimental optical data for 15 inorganic compounds over the electron energy range 500-2000 eV

Correlated vibrational effect

There is no generally applicable yet accurate model for including both anisotropic and correlated thermal vibrational effects in multiple scattering calculations. We here follow Kaduwela, Friedman and Fadley and adopt a correlated vibrational factor which is expected to depend on the distance between the present scatterer and the previous scatterer. With the definition of the effective mean

square displacement with thermal averaging, the equivalent correlated Debye-Waller-type attenuation factor is given by

$$W_c = \exp[-k^2(1-\cos\delta)\sigma_c^2] \tag{31}$$

where σ_c^2 is the mean square relative displacement (MSRD),

$$\begin{aligned}\sigma_c^2 = {} & \frac{3(h/2\pi)^2}{2M_s k_B \theta_D}\left\{1+\frac{4}{t^2}\left[\frac{\pi^2}{6}-\sum_{n=1}^{\infty}\left(\frac{1}{n^2}+\frac{t}{n}\right)\exp(-nt)\right]-\right. \\ & \frac{2}{q_D^2|R_j-R_{j-1}|^2}[1-\cos(q_D|R_j-R_{j-1}|)]+ \\ & \frac{4}{t\,q_D|R_j-R_{j-1}|}\sum_{n=1}^{\infty}\frac{1}{n^2+(q_D|R_j-R_{j-1}|/t)^2}\times \\ & \left.\left[\exp(-nt)(n\sin(q_D|R_j-R_{j-1}|)+\frac{q_D|R_j-R_{j-1}|}{t}\cos(q_D|R_j-R_{j-1}|))-\frac{q_D|R_j-R_{j-1}|}{t}\right]\right\}\end{aligned} \tag{32}$$

where M_s is the substrate or *average atom* atomic mass, k_B is the Boltzmann constant, θ_D is the effective or average atom Debye temperature, $t=\theta_D/T$, T is the sample temperature in K, $|R_j-R_{j-1}|$ is the internuclear distance of the bond vector leading from the site in a single scattering process, i.e. a' in Figure 3, $q_D=\omega_D/v$ is the associated Debye wave vector, v is the velocity of sound which is taken as constant in the Debye approximation, ω_D is the cutoff frequency determined by $(6\theta^2v^3N/V)^{1/3}$, N is the number of acoustic phonon modes in volume V with the wave vector less than q_D, which equals the number of primitive cells, or usually the number of atoms in volume V. So we have

$$q_D = (6\pi^2 N/V)^{1/3} \tag{33}$$

This Debye-Waller attenuation factor has been adopted in equation (32). Accounting for the surface atomic vibration is not as straightforward. The relation between the MSRD and different atomic masses has been given by Allen, Alldredge and Wette,

$$\begin{aligned}\sigma_i^2\sqrt{M_i} &= \sigma_j^2\sqrt{M_j} && (for\ T=0K) \\ \sigma_i^2 &= \sigma_j^2 && (for\ T\to\infty)\end{aligned} \tag{34}$$

Correlating eqn (34) with (32), an effective surface atomic mass is introduced such that

$$\sigma_{bulk}^2\sqrt{M_{bulk}} = \sigma_{surface}^2\sqrt{M_{surface-effective}} \tag{35}$$

where $M_{surface\text{-}effective}=M_{surface}$ if $T/\theta_D<<1$ or $M_{surface\text{-}effective}=M_{bulk}$ if $T/\theta_D>1$. For $T/\theta_D \sim 1$, $M_{surface\text{-}effective}$ is allowed to vary between the surface and bulk atomic masses.

Inner potential correction

A photoelectron with energy E within the jellium medium will have energy $E-E_0$ in the vacuum far from the surface. This loss of kinetic energy E_0 may be related to a potential barrier whose total height is V_0, defined as inner potential. The only energy relevant for the scattering problem is the electron's kinetic energy when it encounters a scattering potential. In photoemission, the scattered electron is detected, and the inner potential represents the physical kinetic energy lost when the electron travels from the scattering potential edge to the detector. This inner potential is

approximately the sum of the work function and the valence band-width.

The inner potential barrier will alter the photoelectron path. We adopt a planar step barrier of height V_0 just outside the last row of ion cores. This is the usual first-order model for the surface barrier, introduced for both low energy photoemission and low-energy electron diffraction. The important consequence of this model is a prediction that the emerging photoelectron will be refracted in a direction away from the surface normal in the manner of optical paths with

$$E_{in} = E_{out} + V_0 \qquad k_{in} \sin \theta_{in} = k_{out} \sin \theta_{out} \tag{36}$$

where V_0 is the inner potential, E_{in} and E_{out} are the electron kinetic energy inside and outside the sample surface, k_{in} and k_{out} the wave number inside and outside the surface, θ_{in} and θ_{out} the photoelectron directions before and after refraction at the surface away from the normal direction, see Figure 4. Substituting eqn (29) into eqn (36), we obtain the inner potential correction formula

$$k_{in} = \sqrt{k_{out}^2 + 0.262483\, V_0} \qquad \theta_{in} = \sin^{-1}\left(\frac{k_{out}}{k_{in}} \sin \theta_{out}\right) \tag{37}$$

Presently there exist no generally acknowledged theoretical or experimental inner potential data for any kind of element. Hence it is treated as an adjustable parameter and fit to experiment.

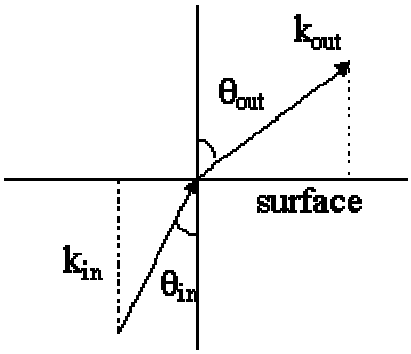


Figure 4. Inner potential correction.

Instrumental angular averaging

The experimental apparatus for measuring photoemission intensities has a small but finite angular resolution characterized by half the angle subtended by the aperture at the source, θ in Figure 5. For small apertures, θ is the radius of the aperture projected on a unit sphere so that the detected area is θ^2. The instrumental angular averaging due to this finite aperture of the detector is done by summing the photoelectron intensities over a grid of points on a circular aperture centered on the nominal emission direction as defined by wave vector **k**. We calculate photoemission intensities I_a, I_b, I_c, I_d and I_e for five different directions (0,0) (θ,0) (θ,θ/2) (θ,θ) (θ,3θ/2) and (θ,2θ), see Figure 5, and assume that the average intensity in each sector is the average of the intensities at the three triangle corners. Then, for example, in sector abc, the average intensity will be $I_{abc}=(I_a+I_b+I_c)/3$. For five point calculations, we obtain the average intensity for a finite aperture of the detector

$$I_{abc} = (2I_a + I_b + I_c + I_d + I_e)/6 \tag{38}$$

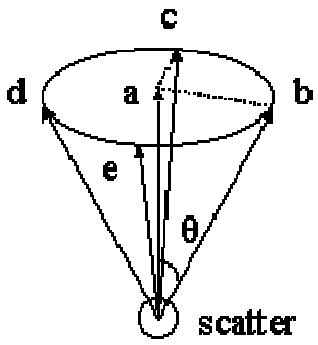


Figure 5. Instrumental angular averaging; θ is the half aperture angle.

Convergence of the Rehr-Albers Approximation

There are several non-structural parameters that need to be taken to convergence when using the R-A representation in photoelectron diffraction simulations: the cluster size $n_{cluster}$, the multiple-scattering order n_{max}, the R-A order $|\mu|_{max}$, and the maximum angular momentum l_{max}. In order to carry out such converged calculations most efficiently, we also include or exclude paths based on their relative importance through a control parameter *pathcut*, as explained in more detail below. All of these parameters except the R-A order also occur in the exact cluster methods. In this section, we address the importance of each of these parameters through a series of calculations. By default, the following quantities were chosen for all the calculations, with variations as specified in subsequent subsections.

The calculations are performed for a default cluster of 86 atoms representing the ideal clean Cu(111) surface, using a lattice constant of 3.615 Å. Intensities are thus summed over emitters in various layers inward from the surface. The cluster shape is chosen to be a half ellipsoid, as shown in Fig. 6, with r = 7 Å and h = 8 Å and the emitter in each layer being positioned as close as possible to the lateral center of it. Scattering phase shifts and radial matrix elements are calculated from a muffin-tin potential due to Moruzzi et al, which uses a 1.26 Å muffin-tin radius; this leads to $l_{max} \approx 7$ and 13 for electron kinetic energies of 100 eV and 400 eV, respectively. The inelastic attenuation length is calculated using the TPP-2M formula of Tanuma, Powell, and Penn, which yields about 4.4 Å at an energy of 100 eV and 8.5 Å at 400 eV. No thermal vibrational damping effect is included. The inner potential was assumed to be 0 eV for these model calculations, thus neglecting any photoelectron refraction effects in crossing the surface barrier. Of course, in any actual comparisons with experiment, this parameter should be set to some reasonable non-zero value. A linearly polarized light source illuminates the surface along a [110] azimuth and at a grazing incidence angle of 10° (i.e. 80° from normal) so that the polarization lies nearly along the surface normal. Photoemission signals are taken from the Cu-3p initial state. By default, we set $n_{max} = 8$, the R-A order = 4 (15x15 matrices), $l_{max} = 20$, and the *pathcut* (to be defined more quantitatively below) = 0.01. Unless otherwise noted these are the values used in all calculations to systematically study parameter choices.

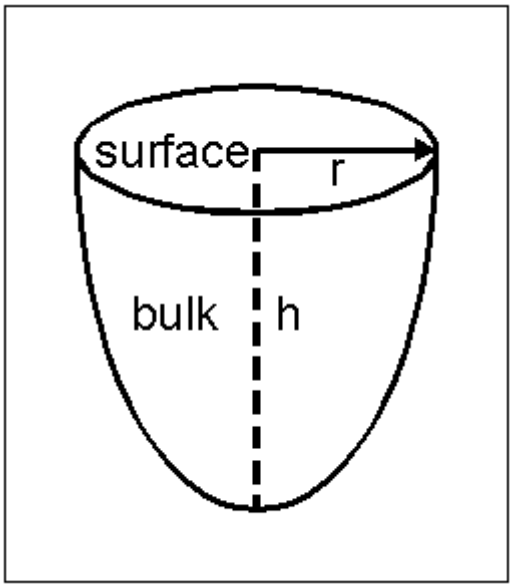


Fig. 6. Cluster shape selection using a half ellipsoid. The side view is a semi-ellipse with minor axis **r** and major axis **h**. The top view is a circle with radius **r**. For the 86-atom cluster used to simulate Cu(111), **h** was 8 Å or 4 emitter layers in depth, and **r** was 7 Å.

Multiple-scattering order n_{max}

In general, photoelectrons are scattered less to large scattering angles than near forward scattering (scattering angle = 0), as illustrated for energies of 156 eV and 547 eV for a single Cu scatterer in Fig. 7. This implies that multiple forward scattering along dense rows of atoms in a crystal can be particularly strong, e.g. for deeper emitters farther from the surface of the cluster. In order to explore the influence of scattering order, the photoelectron detector is thus placed to receive electrons emitted 35.23° off the [111] normal and along a [110] nearest-neighbor forward-scattering direction in the fcc lattice. No angular broadening due to the effective detector aperture was included and kinetic energies of 156 and 547 eV (wave numbers k = 6.4 and 12.0 $Å^{-1}$) were studied.

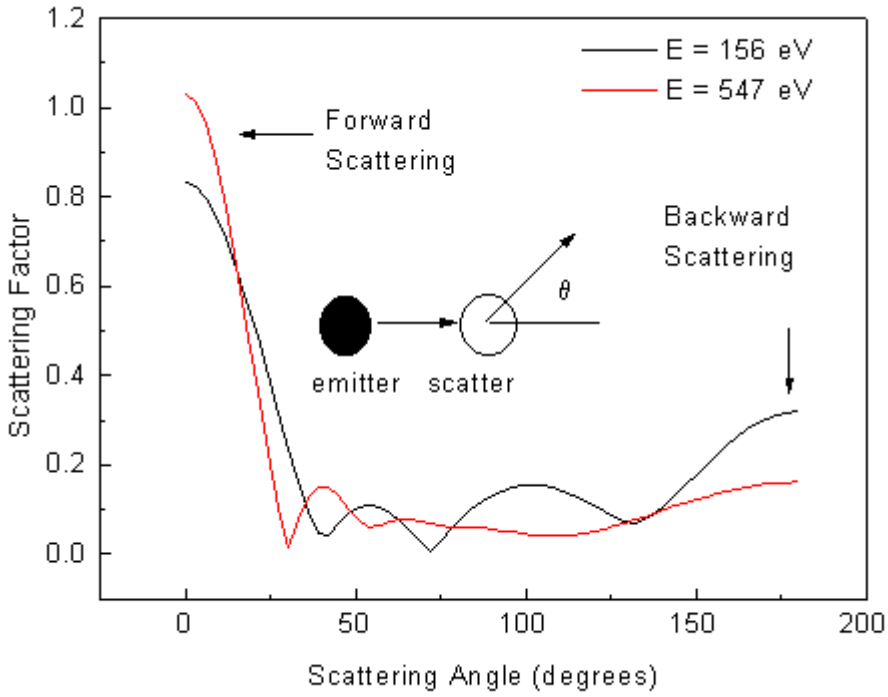


Fig. 7. Cu elastic scattering factors versus scattering angle θ at photoelectron kinetic energies 156 and 547 eV (wave numbers 6.4 and 12.0 $Å^{-1}$). These scattering factors are calculated via the spherical-wave method for a 2-atom cluster with 2.5 Å distance.

Fig. 8 shows the convergence of photoelectron diffraction intensities as a function of multiple scattering orders, for two typical energies, calculated with $(|\mu|+2\nu)_{max} = 4$, i.e. in 4th order of R-A, (15x15) matrix, as compared with exact (i.e. non-R-A) calculations for the same conditions. In these calculations, the emitter is located in the third layer, so that the maximum number of forward scatterers along the direction chosen is two. A photoelectron can thus be scattered once or twice consecutively along the forward-scattering path from emitter to detector. After the 6th order of multiple scattering, the intensities have essentially converged within a few percent, and the 4th-order R-A is essentially identical to the exact results. The same sort of convergence can also be seen in Fig. 9 when the emitter is moved into the 4th layer and the forward-scattering path has three scattering atoms; here, the 7th order appears necessary to insure convergence. No exact results are shown here due to prohibitive computational times with the non-optimized program utilized. Earlier test calculations by Kaduwela et al. with the R-A methods investigated multiple scattering along long straight chains of atoms and showed similar results. The addition of thermal vibration effects should also reduce off-forward scattering amplitudes, thereby aiding the convergence of the multiple scattering series. Finally, any experimental angular averaging will tend to smear out sharper diffraction features associated with longer path length differences, further acting to enhance convergence.

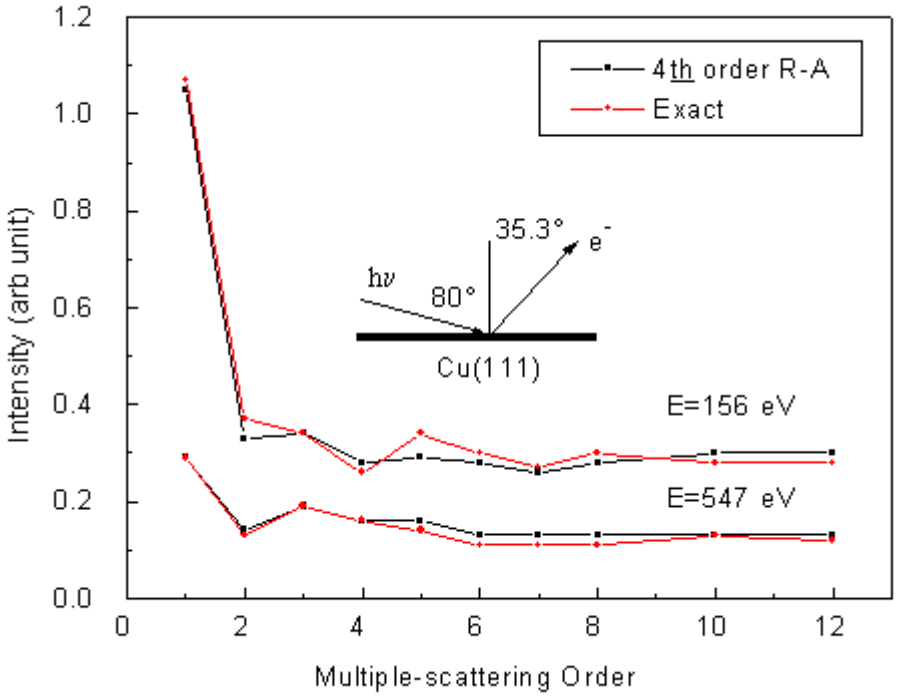


Fig. 8. Calculated Cu 3p photoelectron intensities, as a function of multiple scattering order, from clean Cu(111) in a fixed forward-scattering emission direction, 35.23° off-normal, and for emission from the third layer. A photoelectron can be scattered once or twice consecutively along this forward-scattering path. Photoelectron energies are 156 eV and 547 eV. Default values are used for other parameters. The 4th order R-A is compared with exact calculations.

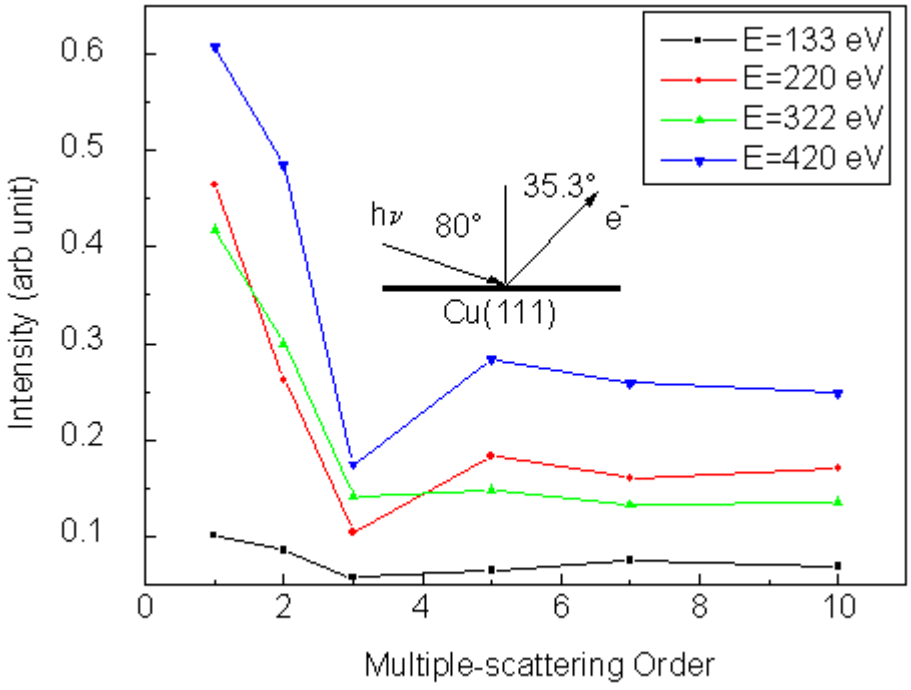


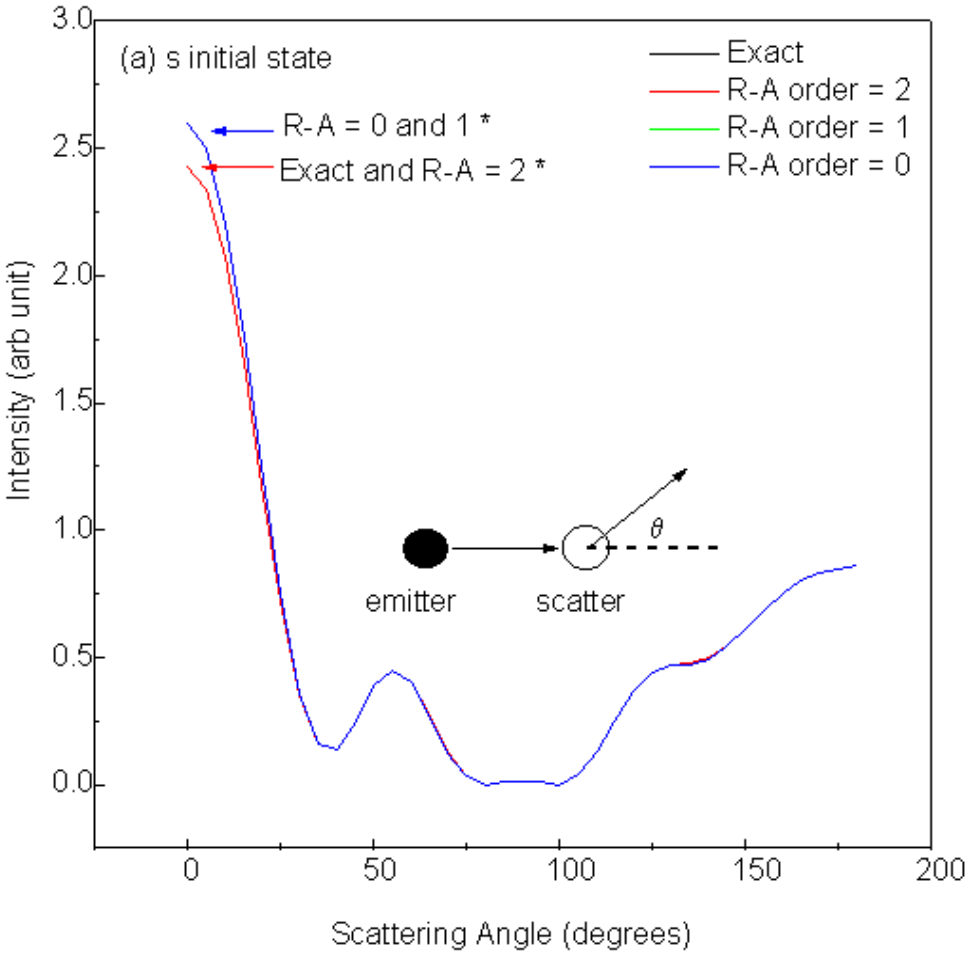
Fig. 9. As Fig. 8, but for emission from the fourth layer. A photoelectron can be scattered up to three times consecutively along this forward-scattering path. Four different photoelectron energies are considered. No exact results were calculated.

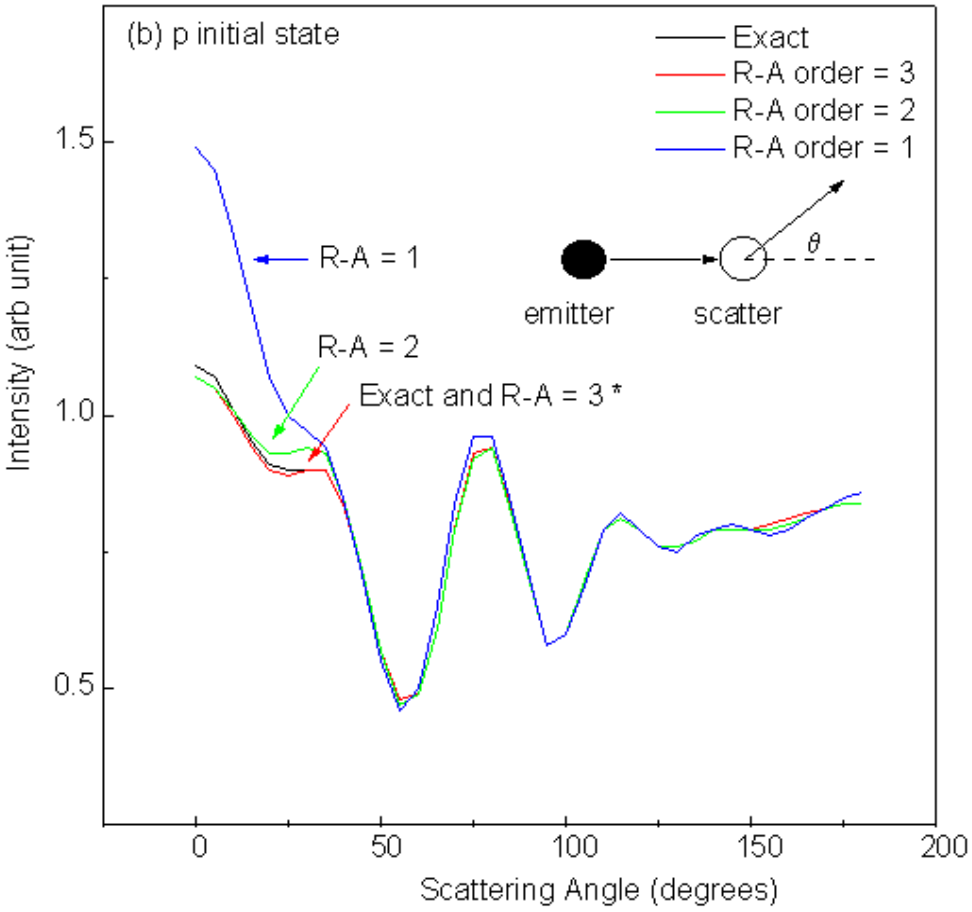
We thus conclude that 4th order R-A is essentially equal to exact, and that the maximum order of scattering needed to adequately simulate PD patterns is 6th or 7th, but probably lower than this with the inclusion of vibrational effects and angular averaging. This agrees with prior studies which have generally concluded that going to 4th or 5th order is sufficient.

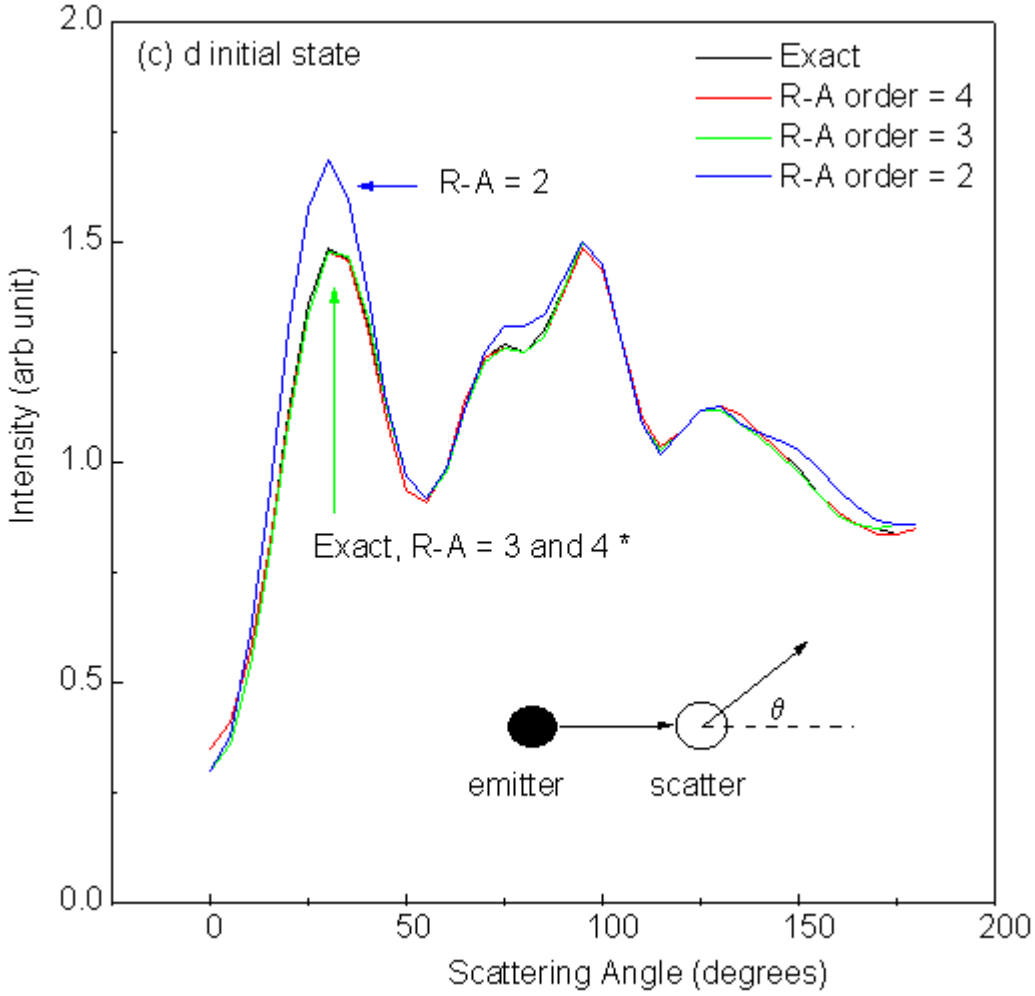
Rehr-Albers approximation order $|\mu|_{max}$ and initial-state effects

To perform a stringent test of the R-A order, we choose a cluster of 2 Cu atoms with an arbitrary small 2.0 Å bond length (somewhat less than the actual 2.56 Å nearest-neighbor distance in Cu), because it provides the maximum sensitivity to different approximation orders. Calculations using a more realistic bond length would converge more rapidly. Closely connected to the required R-A order is the dependence of the R-A approximation on the initial state angular momentum: the 2-atom cluster also provides a good test of this question. A variety of calculations showed that low energies and single scattering are sufficient for this investigation.

Fig. 10 shows scanned-angle photoelectron diffraction intensities for low-energy electrons (E = 61 eV) as a function of scattering angle away from the interatomic axis in the 2-atom cluster. The polarization of the light is here taken to be along the interatomic axis, in order to correspond to those atoms which are in general illuminated most strongly by the primary outgoing wave. Here $\theta = 0°$ means forward scattering, while $\theta = 180°$ corresponds to backward scattering. The four panels in Fig. 10 correspond to excitation from different initial states: s ($l_i = 0$), p ($l_i = 1$), d ($l_i = 2$) and f ($l_i = 3$); and in each panel we compare different R-A orders with exact results under the same conditions. It is seen that for an s initial state emitting into p photoelectron waves, the first R-A order ((3x3) matrices) is adequate and essentially identical to exact. For a p initial state emitting into s and d waves, the second R-A order (6x6) is sufficient. For a d initial state emitting into p and f waves, the third R-A order (10x10) might be needed. And for an f initial state emitting into d and g waves, the fourth R-A order (15x15) is necessary to obtain results accurate within 1%. Here, it was found that changing the initial angular momentum had a strong effect on the diffraction patterns for a two-atom cluster, but the effect of R-A order was not considered. As we shall discuss later, in a larger cluster, subsequent scatterings can be treated with equal or, more frequently lower, order in R-A, so that the R-A order needed for the first scattering is an upper limit for the entire multiple-scattering problem, and does not indicate the real limit on computing time for a given problem.


(a) s initial state
Exact
R-A order = 2
R-A order = 1
R-A order = 0
R-A = 0 and 1 *
Exact and R-A = 2 *
θ
emitter
scatter
Intensity (arb unit)
Scattering Angle (degrees)
3.0
2.5
2.0
1.5
1.0
0.5
0.0
0
50
100
150
200


(b) p initial state
Exact
R-A order = 3
R-A order = 2
R-A order = 1
R-A = 1
R-A = 2
Exact and R-A = 3 *
emitter
scatter
θ
Intensity (arb unit)
1.5
1.0
0.5
0
50
100
150
200
Scattering Angle (degrees)


(c) d initial state
Exact
R-A order = 4
R-A order = 3
R-A order = 2
R-A = 2
Exact, R-A = 3 and 4 *
θ
emitter
scatter
Intensity (arb unit)
Scattering Angle (degrees)
0.0
0.5
1.0
1.5
2.0
0
50
100
150
200

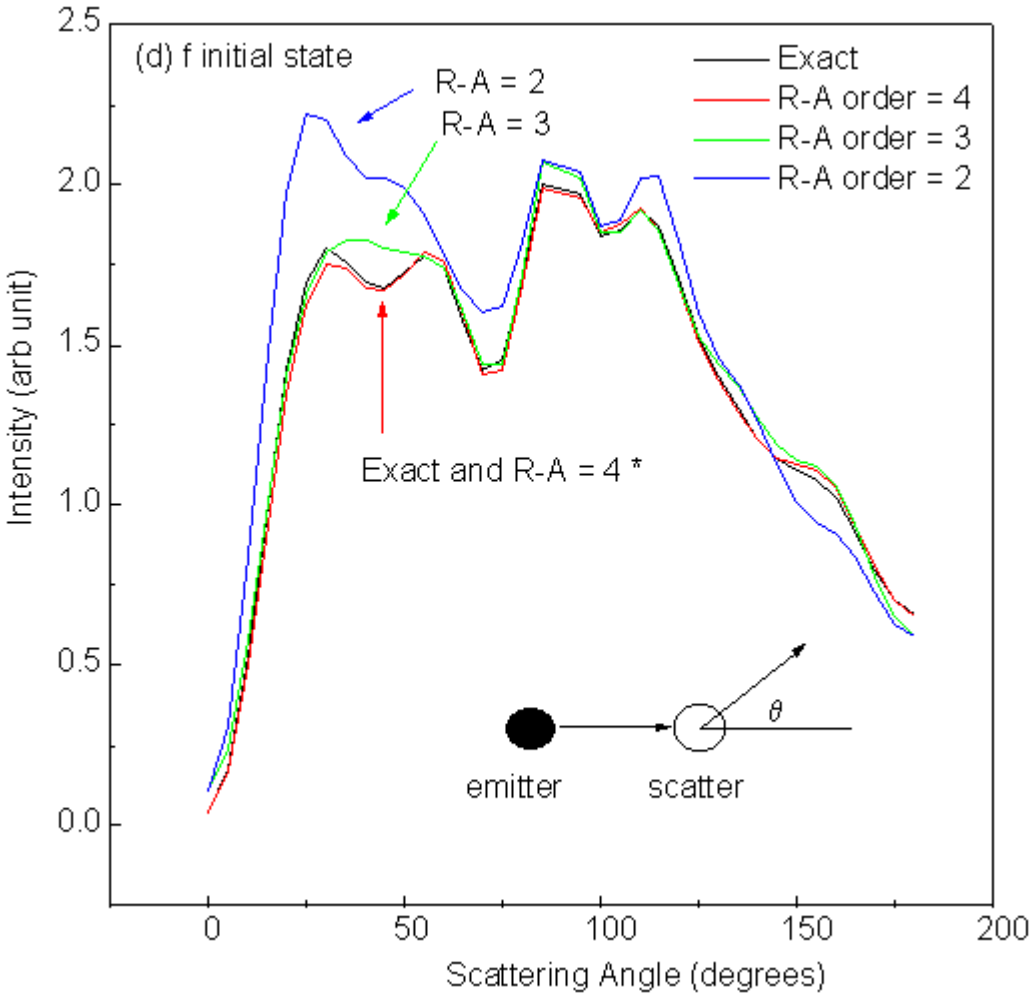


Fig. 10. Single scattering intensities from a 2 Cu-atom cluster with 2.0 Å interatomic spacing and for scattering angles varying between forward (0°) and backward scattering (180°). Emission from four different initial states, s, p, d and f, is considered in panels (a), (b), (c) and (d), respectively. Default values are used for other parameters. Curves labeled with * in each panel are visually identical.

Overall, we thus suggest a simple rule of thumb for guaranteeing adequate results: for emission from an initial state l_i, use the (l_i+1)-th R-A order for the first scattering event after emission. Other subsequent events will generally require lower orders, as dealt with in more detail in the next section.

Pathcut

Our MSCD R-A codes include the ability to neglect multiple-scattering paths that contribute only weakly to the final photoemitted intensities. At the same time, they also allow the R-A order to be adjusted at each stage in a scattering path, a unique feature not utilized before in PD simulations. Both options are controlled by one criterion, called *pathcut*, which is a cutoff criterion with value << 1. In this section, we indicate how this cutoff has been implemented and explore the resulting compromises between time savings and accuracy.

The *pathcut* criterion is applied as follows. Before starting a multiple-scattering calculation, all individual single-center scattering events involving a three-atom *vertex* a → b → c and represented

by a given $F_{\alpha\alpha'}(\rho,\rho')$ of the type shown in Fig. 2 are evaluated separately. The largest value of $F_{00}(\rho,\rho')$ is taken as the reference value. All those elements $F_{\alpha\alpha'}(\rho,\rho')$ that are smaller than a factor *pathcut* times this largest $F_{00}(\rho,\rho')$ are declared to be negligible; this is done dynamically in a multiple-scattering path, so that scattering events further down a path, which are normally weaker because of decay with distance, will be cut off relatively more than early scattering events. In this way, a scattering matrix is automatically reduced as appropriate to a lower-order R-A event with smaller matrix size and faster computation. In particular, if all elements of a matrix are declared negligible, the path is terminated. Because single scattering usually dominates, single scattering paths are calculated using *pathcut* = 0.

To illustrate the effects of *pathcut* for a typical large-cluster case, we show in Table 4 a summary of results for multi-layer emission from an 86-atom Cu(111) cluster at an energy of 100 eV, including the statistical weights in percent of scattering-amplitude matrix sizes when *pathcut* = 0.001 (a typical value that we have found to represent a good compromise between computation time and accuracy). Although second-order multiple scattering requires dealing in about 16% of the cases with 3rd and 4th order R-A or matrices of (10x10) and (15x15) size, for 3rd and higher multiple scattering order, 2nd order R-A and (6x6) matrices are found to be fully adequate. In fact, for 5th or higher multiple scattering order, 1st order R-A is probably adequate. To further quantify the effect of *pathcut* on the quality of the resulting photoemitted intensities, we define an intensity reliability factor R_I as

$$R_I = \sum_i \frac{(I_i(\text{pathcut} = 0) - I_i(\text{pathcut}))^2}{I_i^2(\text{pathcut} = 0) + I_i^2(\text{pathcut})} \tag{39}$$

where I_i represents photoemission intensities, and the sum runs over all the available data points for different energies or angles. Thus $R_I = 0$ represents a perfect calculation, as defined by *pathcut* = 0.

MS order	R-A order (matrix size)					
		0	1	2	3	4
	none	(1x1)	(3x3)	(6x6)	(10x10)	(15x15)
1	0.0	0.0	0.0	0.0	0.0	100.0
2	0.0	1.1	39.3	43.2	7.6	8.7
3	33.6	53.5	12.1	0.7	0.1	0.0
4	27.3	32.3	5.3	0.5	0.0	0.0
5	61.9	20.1	3.0	0.3	0.0	0.0
6	83.8	13.7	2.2	0.3	0.0	0.0
7	87.6	10.3	1.8	0.3	0.0	0.0
8	89.7	8.3	1.7	0.3	0.0	0.0

Table 4. Distribution (in percent) of different scattering amplitude matrix sizes as a function of multiple-scattering order (MS order) for an 86 atom Cu cluster with *pathcut* = 0.001, using default values at energy E = 100 eV. The column labeled *none* represents weak events that terminate a path.

Fig. 11 shows the intensity reliability and calculation time as a function of *pathcut* for an energy scan of Cu 3p intensity over the energy range 60 eV to 550 eV for our 86-atom Cu(111) cluster. It

can be seen that setting a *pathcut* value of 0.001 can easily save an order of magnitude of computation time compared to the full calculation without cut (*pathcut* = 0), and also that a great deal of time gain is achieved even with this small a cutoff criterion. That is, going from 0 to 0.001 gains by a factor of about 19, whereas going from 0 to 0.01 gains by about 75. Fig. 12 gives us a feeling for the quality of curve-to-curve comparisons at different *pathcut* values.

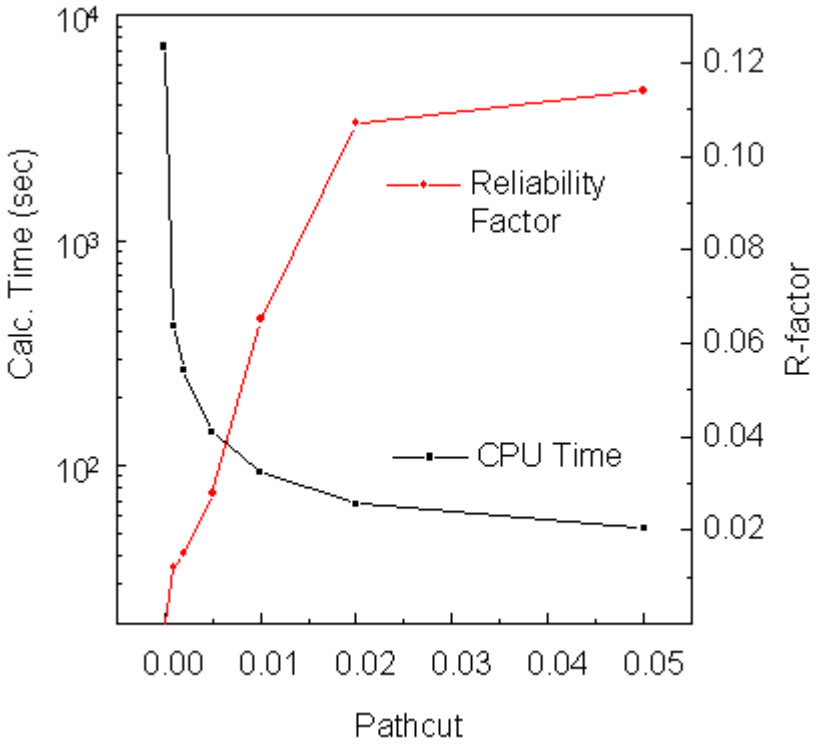


Fig. 11. Intensity reliability factor R_I (Eq. (39)) and calculation time on a 200MHz Sun Sparc Ultra-2 workstation as a function of *pathcut*, for various choices of this parameter in second-order R-A scanned-energy calculations for the 86-atom Cu(111) cluster. A value of 0 for *pathcut* corresponds to inclusion of all scattering events.

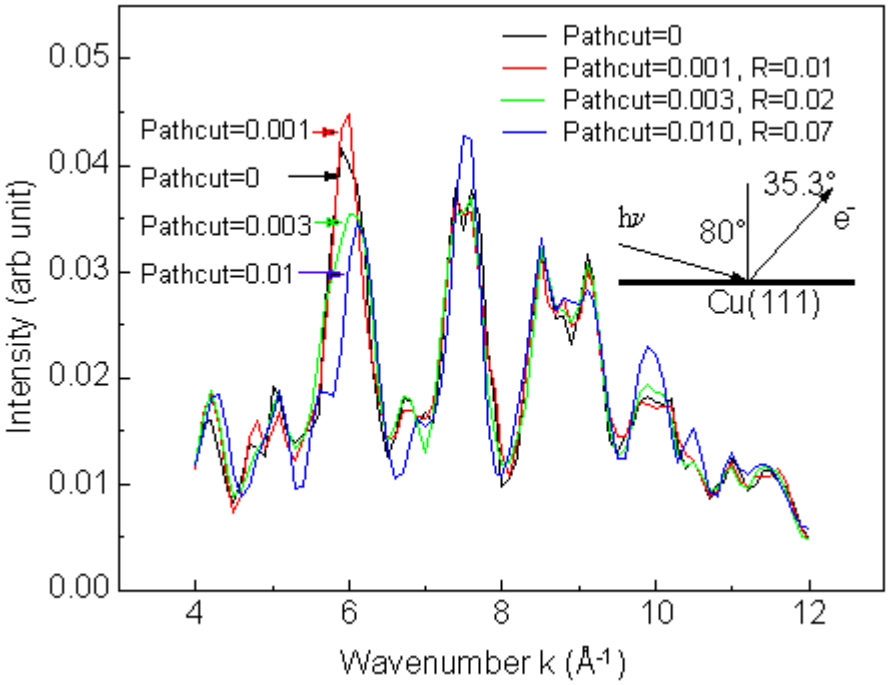


Fig. 12. Curve-to-curve comparisons of scanned-energy calculations for the 86-atom cluster and for different values of *pathcut*. Default values are used for other parameters.

From this analysis and other calculations, we find that a *pathcut* value of 0.001 is fully adequate for the quantitative modeling of photoelectron diffraction data.

Cluster size

If a cluster is to represent an infinitely extended surface and/or include multilayer emission from a bulk specimen (as in the Cu case considered here), its size must be chosen large enough. To properly scale this problem, photoelectron waves leaving an emitter in free space decay in intensity with the inverse square of the distance from the emitter, i.e. as $1/r^2$: if there were no other damping effects, this would require an infinitely large cluster, since the number of scatterers on a shell at a given distance increases with the square of that distance, compensating the $1/r^2$ decay. Inelastic scattering adds an exponential decay factor, described theoretically by the inelastic attenuation length, which ensures that a finite cluster suffices. Vibrational effects and angular broadening act to further shrink the volume that is effective in producing diffraction modulations.

Fig. 13 again shows scanned-energy results for the ideal clean Cu(111), but this time calculated for clusters of different sizes. The photon polarization angle is again 10° off-normal and the intensities are taken from the Cu 3p core level in the direction of normal emission, allowing emission from all layers in the cluster. No *pathcut* is considered. To better compare these scanned-energy curves, we plot the usual $\chi(k)$ curves defined as

$$\chi(k) = (I(k) - I_0(k)) / I_0(k) \qquad (40)$$

where $I(k)$ is the photoemission intensity at wave number k, and $I_0(k)$ is the background subtracted from the intensity vs. wave number curve by using a spline fitting method. From Fig. 13 we can see that a 119-atom cluster yields most peaks and valleys at proper locations. Larger clusters become necessary for finer details. However, in practice, other effects not included here favor the sufficiency of smaller clusters: namely, vibrational damping of diffraction, and the experimental

angular aperture (typically ±3° to ±5°), both of which will tend to smooth out fine structure.

Thus, we conclude that clusters of about 100 atoms in size should be sufficient for most problems, in agreement with prior studies.

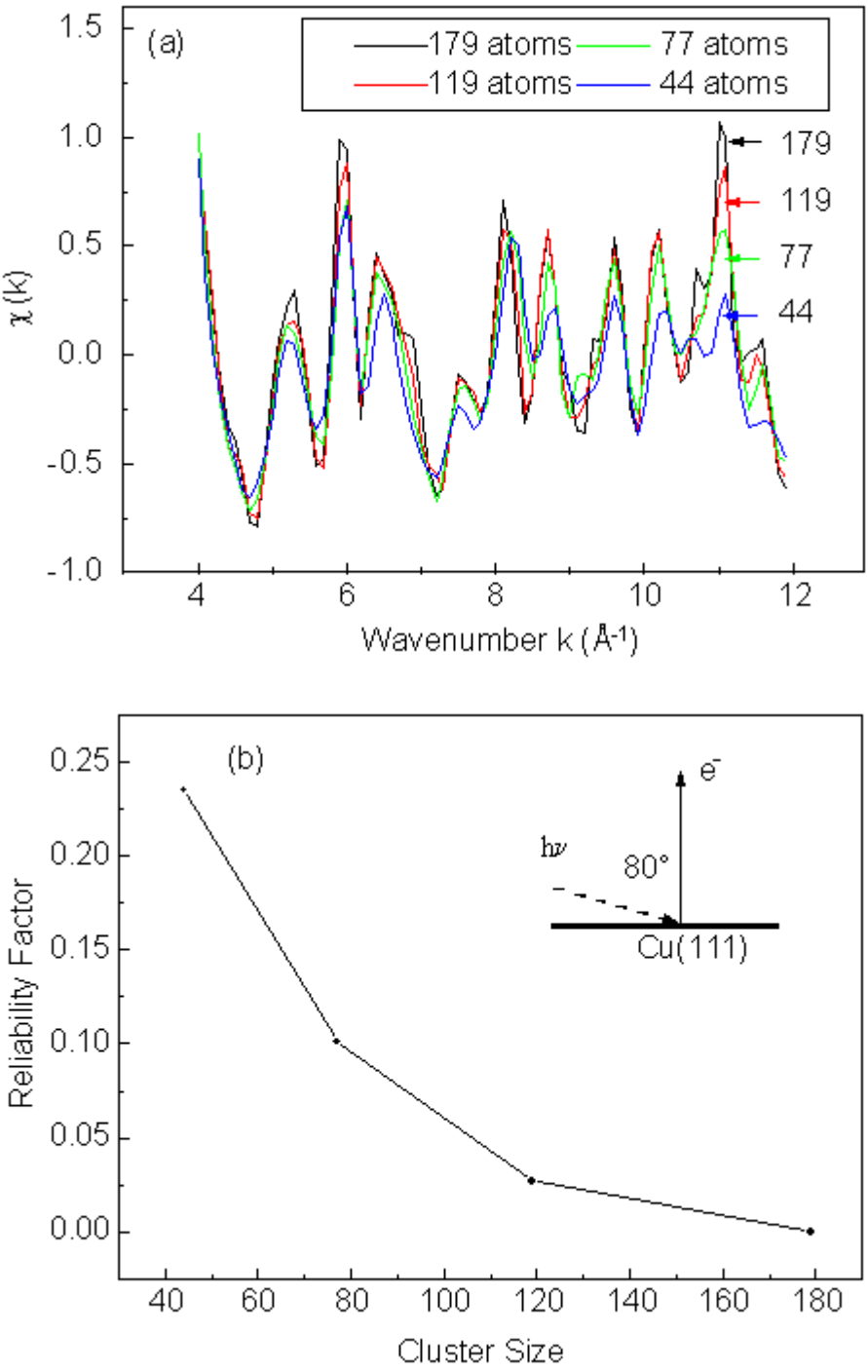


Fig. 13. (a) Effect of cluster size on scanned-energy calculated photoelectron diffraction χ(k) curves for Cu(111), with clusters sizes of 44, 77, 119, and 179 atoms. Default values are used for other parameters. (b) Reliability factor as a function of cluster size, with the 179-atom result used as the reference. See text for further details.

Reliability in structure determinations

Finally, we consider the reliability with which such R-A calculations can be used to determine atomic structures, using the classic approach of theory-experiment comparison via reliability factors or R-factors. Although various definitions of R-factors exist, we will here use a rather straightforward definition of the goodness of fit between theory and experiment for photoelectron diffraction data:

$$R = \sum_i \frac{(\chi_{ci} - \chi_{ei})^2}{\chi_{ci}^2 + \chi_{ei}^2} \qquad (41)$$

where χ_{ci} and χ_{ei} are calculated and experimental χ curves, respectively (c.f. Eq. (39)). As one way of estimating the reliability of the Rehr-Albers approximation for structure determinations, we have replaced the experimental data with the calculated scanned-energy (60 to 550 eV) results from the exact formalism, which is based on a smaller 35-atom cluster representing Cu 3p emission from ideal clean Cu(111). We have used our R-A method with R-A order only up to 2 (6x6 matrices) to explore sensitivity to possible variations in the outermost interlayer spacing by calculating the same scanned-energy curve and quantifying the fit to the exact result. The variation of R with interlayer spacing is shown in Fig. 14. It indicates a best fit R value of 0.009 at precisely the interlayer spacing that was used in the exact calculation, thus giving confidence in the ability of this method, even at 2nd R-A order, to accurately determine structure, while saving computer time. Our current implementation of the R-A method with *pathcut* and adjustable orders should do even better than this.

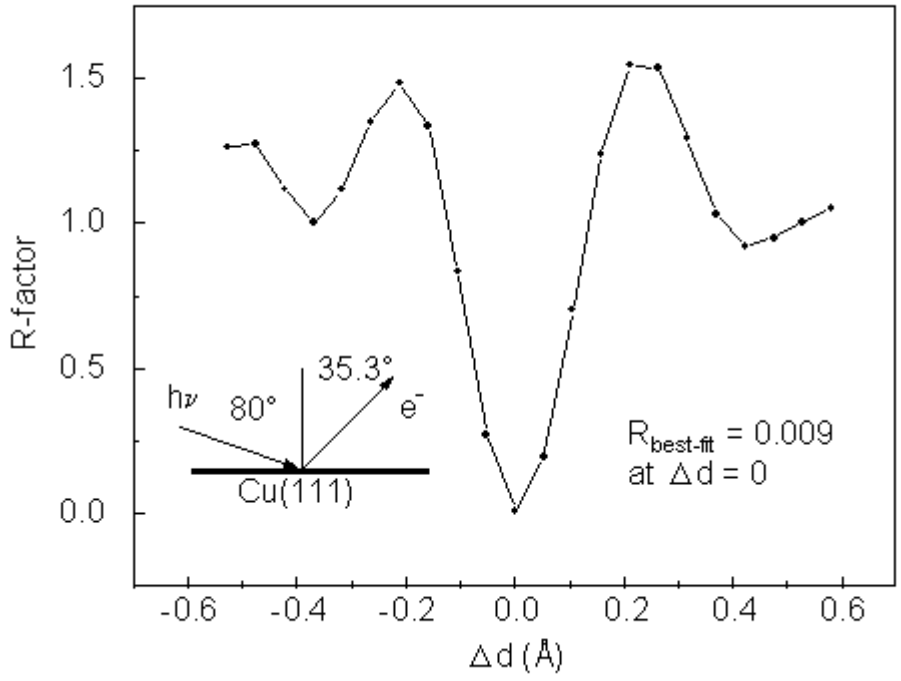


Fig. 14. Assessment of atomic position reliability of photoelectron diffraction calculations using the second-order R-A approximation, shown by varying the first interlayer spacing and comparing to intensities calculated by using the exact formalism with zero interlayer relaxation. A cluster of 35 atoms was used.

Conclusions

The Rehr-Albers (R-A) separable propagator approximation up to 4th order (and using up to (15x15) matrices) has been applied to the calculation of photoelectron diffraction curves. By replacing the propagator matrices in the exact Green's function formalism by the much smaller scattering-amplitude matrices of R-A, this approximation saves much computation time. Our convergence tests for typical conditions in photoelectron diffraction indicate that 4th order R-A is highly accurate for all cases likely to be encountered. Furthermore, 2nd-order R-A (with (6x6) matrices) applied with clusters of 100 or more atoms and at least 7th-order multiple scattering, and using a *pathcut* of about 0.001, provides excellent results within 5% of *exact* results for most cases, particularly if the initial state is of s or p type. Higher Rehr-Albers orders are necessary for the first scattering events involved with initial states of d type (3rd order) and f type (4th order), but can be neglected in later events. We have also implemented R-A in a program which automatically adjusts the R-A order from 4th downward according to the *pathcut* criterion, and this should permit fully quantitative and maximally efficient calculations for any situation. At least an order of magnitude in computation time is saved by recognizing that lower orders of Rehr-Albers suffice for most higher-order multiple scattering events. Larger clusters may be necessary for describing all fine structure in diffraction curves, but approximately 100 atoms should be the maximum needed for most cases. Looking ahead to future applications of the Rehr-Albers method, we note that several simulations of actual experimental data, e.g. on surfaces of W(110), O/W(110), Li/Al(111), and MnO(100), etc., have also been performed using MSCD.

Speeding up the calculation

Iterative summing

To simplify the notations for summing equation (12), we can rewrite it using matrices and arrays,

$$G_{00,lf}^{(n-1)}(R_1, R_2, \cdots, R_n) = \sum_{(paths)} \tilde{\Gamma}(\varrho_n) F(\varrho_n, \varrho_{n-1}) \cdots \times F(\varrho_3, \varrho_2) F(\varrho_2, \varrho_1) \Gamma(\varrho_1) \tag{42}$$

where, for the 2nd order R-A approximation (6x6 matrix size)

$$\tilde{\Gamma}(\varrho_n) = \left[\tilde{\Gamma}_1, \tilde{\Gamma}_2, \tilde{\Gamma}_3, \tilde{\Gamma}_4, \tilde{\Gamma}_5, \tilde{\Gamma}_6 \right]$$

$$F(\varrho_n, \varrho_{n-1}) = \begin{bmatrix} F_{11} & F_{12} & F_{13} & F_{14} & F_{15} & F_{16} \\ F_{21} & F_{22} & F_{23} & F_{24} & F_{25} & F_{26} \\ F_{31} & F_{32} & F_{33} & F_{34} & F_{35} & F_{36} \\ F_{41} & F_{42} & F_{43} & F_{44} & F_{45} & F_{46} \\ F_{51} & F_{52} & F_{53} & F_{54} & F_{55} & F_{56} \\ F_{61} & F_{62} & F_{63} & F_{64} & F_{65} & F_{66} \end{bmatrix} \qquad \Gamma(\varrho_1) = \begin{bmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \\ \Gamma_4 \\ \Gamma_5 \\ \Gamma_6 \end{bmatrix} \tag{43}$$

The relationship between the element subscripts and Rehr-Albers expansion indices [$\lambda' = (\mu'\ \nu')$, $\lambda = (\mu\ \nu)$] is listed in Tables 5 - 8. Because the F elements converge for large ρ roughly as $(\rho')^{-(|\mu'|+2\nu')}(\rho)^{-(|\mu|+2\nu)}$, we can easily see, in this notation, there are only one element in each above array or matrix ($\tilde{\tilde{\Gamma}}_1$, Γ_1 and F_{11}) for zeroth Rehr-Albers expansion order, equivalent to the asymptotic high-energy limit and to the point-scattering approximation. We only need to keep the first three elements in the $\tilde{\tilde{\Gamma}}$ and Γ arrays and a (3x3) matrix in F up to first Rehr-Albers expansion order. For the full calculations, Rehr and Albers have demonstrated that in practice it seems sufficient to retain terms only to second order, i.e. the (6x6) matrix, in most cases.

	$\tilde{\tilde{\Gamma}}_1$	$\tilde{\tilde{\Gamma}}_2$	$\tilde{\tilde{\Gamma}}_3$	$\tilde{\tilde{\Gamma}}_4$	$\tilde{\tilde{\Gamma}}_5$	$\tilde{\tilde{\Gamma}}_6$
$\lambda' = (\mu', \nu')$	(0,0)	(1,0)	(-1,0)	(0,1)	(2,0)	(-2,0)
$L_n=(l,m)$	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)	(0,0)

Table 5. The relationship between the subscripts of the $\tilde{\tilde{\Gamma}}$ factor elements and the Rehr-Albers expansion indices.

	F_{1x}	F_{2x}	F_{3x}	F_{4x}	F_{5x}	F_{6x}
$\lambda' = (\mu',\nu')$	(0,0)	(1,0)	(-1,0)	(0,1)	(2,0)	(-2,0)

Table 6. The relationship between the subscripts of the scattering amplitude matrix elements and the Rehr-Albers expansion indices.

	F_{x1}	F_{x2}	F_{x3}	F_{x4}	F_{x5}	F_{x6}
$\lambda = (\mu,\nu)$	(0,0)	(1,0)	(-1,0)	(0,1)	(2,0)	(-2,0)

Table 7. The relationship between the subscripts of the scattering amplitude matrix elements and the Rehr-Albers expansion indices.

	Γ_1	Γ_2	Γ_3	Γ_4	Γ_5	Γ_6
$\lambda = (\mu,\nu)$	(0,0)	(1,0)	(-1,0)	(0,1)	(2,0)	(-2,0)
$L_f=(l,m)$	(l_f,m_f)	(l_f,m_f)	(l_f,m_f)	(l_f,m_f)	(l_f,m_f)	(l_f,m_f)

Table 8. The relationship between the subscripts of the Γ factor elements and the Rehr-Albers expansion indices.

From equation (42), we see that the number of array or matrix multiplications dominates the total calculation time. For the (n-1)-th scattering order term, the number of terms in the summation is proportional to N^{n-2} times the number of emitters, and times the number of $L_f=(l_f,m_f)$ combinations, where N is the number of atoms in question. This procedure takes much time, even when using

pathcut.

We have therefore introduced an iterative summing method, which makes the calculation time be proportional to N^3. For the commonly used 6th to 8th scattering order, such a method can cut down the computation time enormously. To that end, we rewrite the equation (42) into a forward summing version as

$$G_{00,\underline{\rho}}^{(n-1)}(R_1,R_2,\cdots,R_n)=\sum_{\underline{\rho}n}[\tilde{\Gamma}(\underline{\rho}_n)\sum_{\underline{\rho}n-1}[F(\underline{\rho}_n,\underline{\rho}_{n-1})\times\sum_{\underline{\rho}n-2}[F(\underline{\rho}_{n-1},\underline{\rho}_{n-2})\cdots\times \sum_{\underline{\rho}2}[F(\underline{\rho}_3,\underline{\rho}_2)\times\sum_{\underline{\rho}1}[F(\underline{\rho}_2,\underline{\rho}_1)\times\Gamma(\underline{\rho}_1)]]\cdots]]] \tag{44}$$

or in a backward summing version as

$$G_{00,\underline{\rho}}^{(n-1)}(R_1,R_2,\cdots,R_n)=\sum_{\underline{\rho}1}[\sum_{\underline{\rho}2}[\sum_{\underline{\rho}3}[\cdots\sum_{\underline{\rho}n-1}[\sum_{\underline{\rho}n}\tilde{\Gamma}(\underline{\rho}_n)F(\underline{\rho}_n,\underline{\rho}_{n-1})]\times F(\underline{\rho}_{n-1},\underline{\rho}_{n-2})]\cdots\times F(\underline{\rho}_3,\underline{\rho}_2)]\times F(\underline{\rho}_2,\underline{\rho}_1)]\times\Gamma(\underline{\rho}_1) \tag{45}$$

The calculation time will be proportional to $N_k \cdot N_{mi} \cdot (msorder-2) \cdot N_a^3$ for forward summing and $N_k \cdot N_d \cdot (msorder-2) \cdot N_a^3$ for backward summing, where N_k is the number of photoelectron energies to be simulated, $N_{mi} = (2l_i+1)$ the number of possible magnetic quantum numbers, N_d the number of photoelectron exit angles, msorder the multiple scattering order, and N_a the number of atoms in the cluster.

Searching the symmetries

The most time-saving contribution which Rehr and Albers made is the derivation of the separable representation of the scattering event, i.e. equation (26). Let's rewrite it as

$$F_{\lambda\lambda'}(\underline{\rho},\underline{\rho}')=\exp(-i\mu'\alpha)f_{\lambda\lambda'}(\underline{\rho},\underline{\rho}',\beta)\exp(-i\mu'\gamma) \tag{46}$$

Thus, the energy-dependent scattering factor $f_{\lambda\lambda'}(\boldsymbol{\rho},\boldsymbol{\rho}',\beta)$ now depends only on the bond lengths and the scattering angle β (which is equivalent to $\arccos(\boldsymbol{\rho},\boldsymbol{\rho}')$),

$$f_{\lambda\lambda'}(\underline{\rho},\underline{\rho}',\beta)=\exp(-\frac{a'}{2\lambda}-k^2(1-\cos\beta)\sigma_c^2)\times \frac{\exp(i\underline{\rho}')}{\underline{\rho}'}\sum_l t_l\gamma_{\mu\nu}^l(\underline{\rho})d_{\mu\mu}^l(\beta)\tilde{\gamma}_{\mu\nu}^l(\underline{\rho}') \tag{47}$$

This form considerably reduces the number of independent computations of the scattering matrices. Furthermore the only additional angle needed for each element in the calculation is the combinations of α and γ at each scattering site.

The most time-consuming computation in each path is that of equation (47). There are N^3 choices of $\boldsymbol{\rho}$ and $\boldsymbol{\rho}'$, where N is the number of atoms in the cluster. That means we have to compute N^3 scattering events $f_{\lambda\lambda'}(\boldsymbol{\rho},\boldsymbol{\rho}',\beta)$. However, in practice, in a crystalline surface with its periodic and other symmetries, many events $f_{\lambda\lambda'}$ have the same $\boldsymbol{\rho}$, $\boldsymbol{\rho}'$ and β values (the absolute orientation of the vectors $\boldsymbol{\rho}$, $\boldsymbol{\rho}'$ does not matter). Therefore, there will be many identical results for $f_{\lambda\lambda'}(\boldsymbol{\rho},\boldsymbol{\rho}',\beta)$. So, we can investigate all the scattering events, and calculate only once the identical events. In our

program, we further pre-calculate $\tilde{r}(\rho)$, $r(\rho)$ and $r(\beta)$ for a preselected series of **ρ** and β values, respectively, then use interpolation to produce their values in the computation of $f_{\lambda\lambda'}$ (**ρ,ρ'**,β).

Applications of the MSCD package

The following case studies are extracts from published papers

---

Calculation and fitting of Ni(001)+c(2x2)-S/S1s

We performed a simultaneous fitting for both normal and off-normal photoelectron diffraction energy scanning curves taken from the Sulfur 1s core-level initial states of Ni(001)+c(2x2)-S. These two ARPEFS measurements were obtained on beamline 3-3, a soft x-ray double crystal monochromator, at Stanford Synchrotron Radiation Laboratory by Barton et al. at room temperature. The first one is a normal emission curve, which used normal emission with the polarization vector inclined 30° from the normal in a [100] direction. The second is an off-normal emission curve, with both emission and polarization vectors aligned with a bulk [011] axis, making an angle of 45° with the surface normal. The instrumental aperture half-angle is about 3°. The Ni(001) substrate has an fcc structure with lattice constant 3.52 Å. The overlayer Sulfur atom occupies a four-fold hollow adsorption site.

Both curves are fit simultaneously with a 83-atom cluster using the three step fitting process (net search, Simplex Downhill, and Marquardt methods). The multiple scattering order is set to 8, the Rehr-Albers approximation order set to 2, and pathcut is set to 0.01. We chose four fitting parameters: the inner potential, the Debye temperature, the spacing between first and second (S-Ni) layers, and the spacing between the second and third (Ni-Ni) layers. Figures 15 and 16 show the best-fit calculated curves (dashed lines) for both experimental curves (solid lines). The fitting procedure gives an inner potential of 10.65 eV, a Debye temperature of 430 K, a spacing between first and second (S-Ni) layers of 1.31 Å, and a spacing between second and third (Ni-Ni) layers of 1.84 Å.

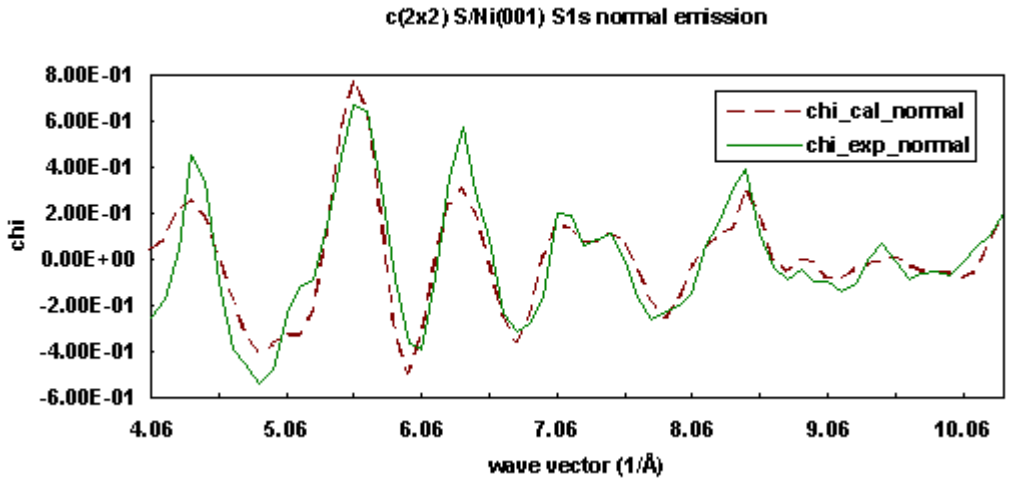


Fig. 15. Numerical simulation (dashed line) of experimental photoelectron diffraction scanned energy curve (solid line) for Ni(001)+c(2x2)-S/S1s in the surface normal exit direction.

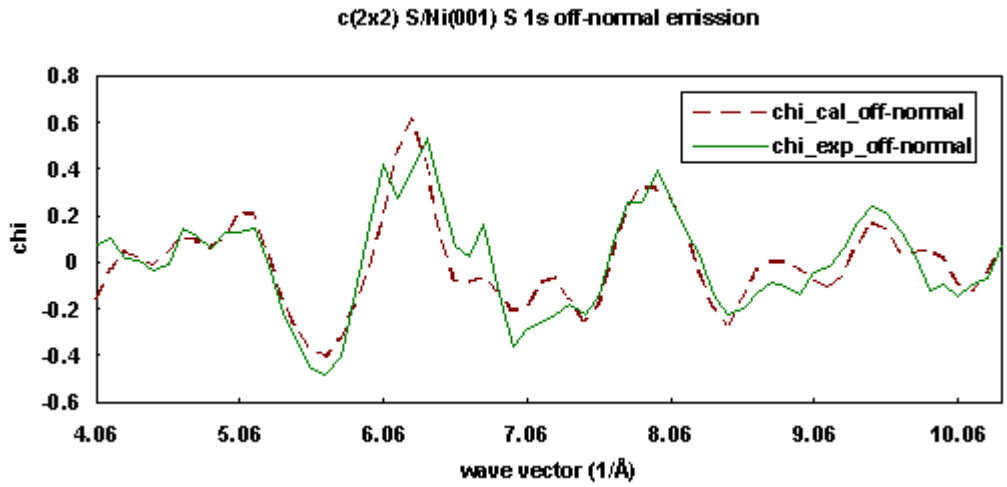


Fig. 16. Numerical simulation (dashed line) of the experimental photoelectron diffraction scanned energy curve (solid line) for Ni(001)+c(2x2)-S/S1s in 45° off-normal exit direction.

Calculation and fitting for Fe(001)+c(2x2)-P/P1s

The experiments were performed in an ultra-high vacuum chamber using beamline 3-3, a soft x-ray double crystal monochromator, at Stanford Synchrotron Radiation Laboratory. The photoemission data were collected from the P 1s core level of Fe(001)+c(2x2)-P in two different experimental geometries at room temperature. In the first data set, the photoemission angle was normal to the Fe(001) surface, and the photon polarization vector was 35° from the surface normal. This geometry gives information which is most sensitive to the Fe atoms directly below the P atoms. The second set of photoemission data was collected along the [011] direction, i.e. 45° off normal toward the (011) crystallographic plane, and the photon polarization vector was oriented parallel to the emission angle. By taking ARPEFS data off normal, the structure sensitivity parallel to the surface is enhanced. Analyzed together, the two different experimental geometries allow for an accurate determination of interlayer spacings, bond lengths, and bond angles.

The Fe substrate has a bcc structure with lattice constant 2.87 Å. A simultaneous fitting technique indicates that the P atoms adsorb in the high-coordination four-fold hollow sites. Figures 17 and 18 show the comparison of calculated curves (dashed lines) and experimental curves (solid lines). The P atoms were determined to bond 1.02 Å above the first layer of Fe atoms. The Fe-P-Fe bond angle is thus 140.6°. Assuming the radius of the Fe atoms to be 1.24 Å, the effective P radius is 1.03 Å. The inner potential was found tobe 15.0 eV. It was also determined that there is no relaxation of the first or second Fe-Fe interlayer spacings from the bulk value of 1.43 Å. To test this fitting method, each data set was fit individually and these results were in good structural agreement.

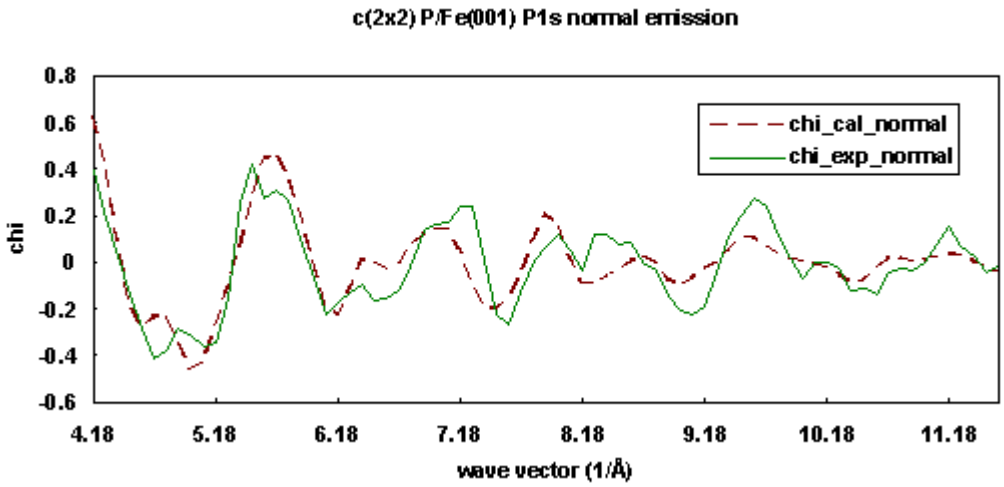


Fig. 17. Numerical simulation (dashed line) of the experimental photoelectron diffraction scanned energy curve (solid line) for Fe(001)+c(2x2)-P/P1s in the surface normal exit direction.

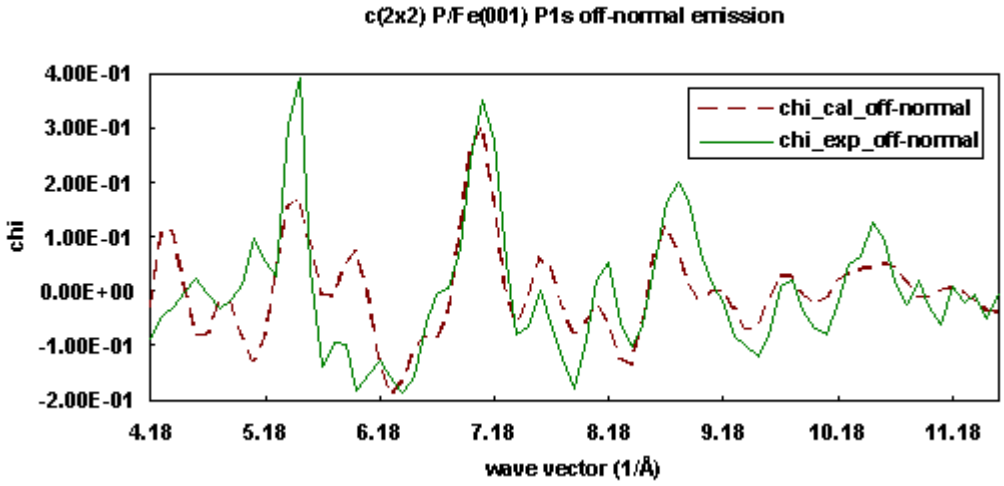


Fig. 18. Numerical simulation (dashed line) of the experimental photoelectron diffraction scanned energy curve (solid line) for Fe(001)+c(2x2)-P/P1s in 45° off-normal exit direction.

Calculation and fitting for clean Cu(111)-3s and 3p

The experiments were performed using the Advanced Light Source at Lawrence Berkeley National Laboratory on beamline 9.3.2 with a soft x-ray spherical grating monochromator. The data were collected using an angle-resolving electrostatic hemispherical electron energy analyzer which is rotatable 360° around the sample's vertical axis and 100° around the sample's horizontal axis. The angle of incidence of the light on the crystal was oriented 80° from the surface normal. The photon polarization vector was thus oriented 10° from the surface normal. The crystal was cooled to about 80K throughout the data collection. Photoemission data were taken from the clean Cu(111) 3p core level and subsequently the 3s core level. The two data sets were acquired in normal exit direction within a few hours of each other.

The fitting procedures were appied to a 77-atom cluster Cu(111) surface separately for the 3s and 3p initial states. The R-factor was minimized as a function of emission angle θ_e and ϕ_e. For the Cu(111) 3s fitting, the R-factor minimum is rather shallow in the range $0° < \theta_e < 5°$. However, for

$\theta_e > 5°$, the R-factor rises sharply. In contrast, the Cu 3p R-factor minimum is very steep away from the minimum at $\theta_e = 5°$. This result indicates that the detected intensity distribution of Cu 3s photoemission is less directional than that of Cu 3p photoemission. Because the emission angle difference of 1° is so important, great care must be taken during the alignment of the experimental system, and also, the modeling must search angle-space to finally obtain the optimum fit to the data. In this case, the emission direction was optimized at 5° off-normal toward the [111] direction. The fitting process for each curve determined that the spacing between first and second layer is 2.06 Å, slightly contracted from the bulk value, 2.09 Å, which agrees with the previous LEED studies. Figures 19 and 20 show the experimental data (solid lines) and their best-fit simulation (dashed lines).

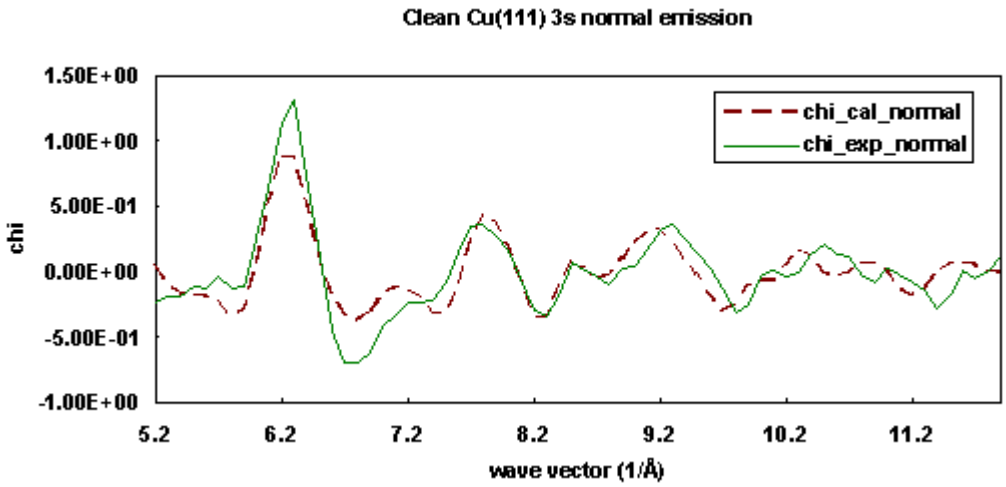


Fig. 19. Numerical simulation (dashed line) of the experimental photoemission diffraction scanned energy curve (solid line) for clean Cu(111) 3s in the surface normal exit direction.

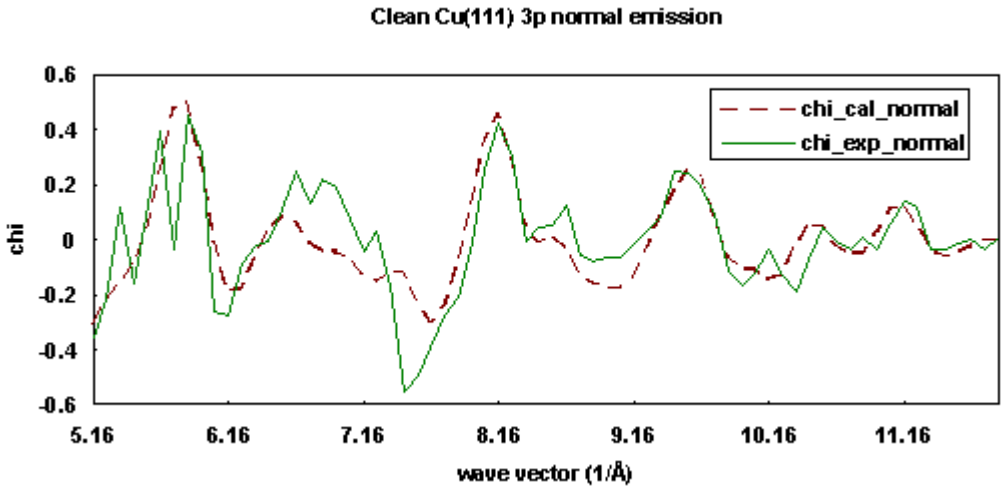


Fig. 20. Numerical simulation (dashed line) of the experimental photoemission diffraction scanned energy curve (solid line) for clean Cu(111) 3p in the surface normal exit direction.

Calculation and fitting for clean Ni(111)-3p

The experiment was performed at the National Synchrotron Light Source at Brookhaven National Laboratory on beamline U3-C, a soft x-ray beamline with a five meter extended range grasshopper

monochromator having a fixed exit geometry. The angle of incidence of the light on the crystal was 55° from the surface normal away from the crystal (011) plane. The photon polarization vector was thus oriented 35° from the surface normal and perpendicular to the crystal (011) plane. The analyzer was oriented normal to the Ni(111) surface and the crystal was cooled to about 100K throughout the data collection.

The fitting was done with a 74-atom cluster representing the clean Ni(111) surface. The R-factor was minimized as a function of emission angle θ_e and ϕ_e, which gave an optimized emission direction of 5° off normal. The spacing between the first two layers was found to be 2.06 Å, a slight expansion of the bulk value, 2.03 Å. Figure 21 campares the experimental curve (solid line) and its best fit simulation (dashed line).

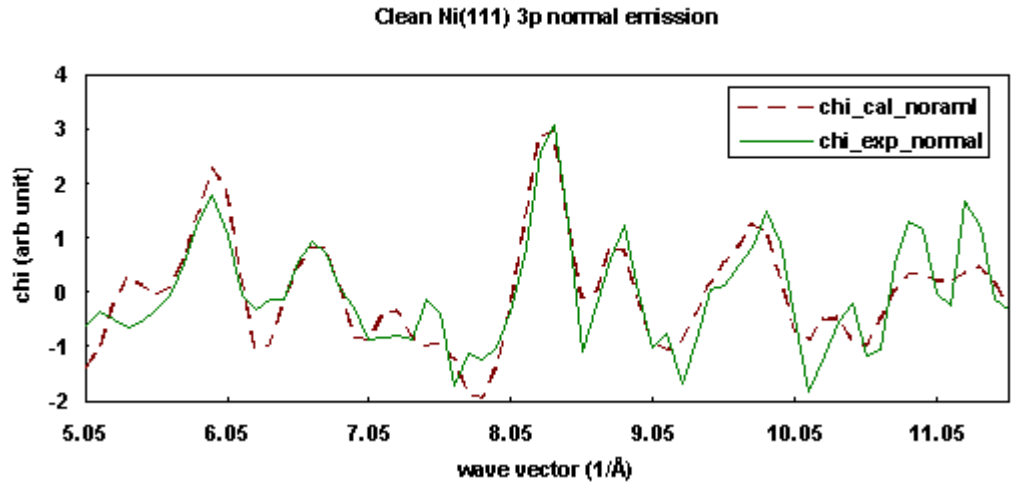


Fig. 21. Numerical simulation (dashed line) of the experimental photoemission diffraction scanned energy curve (solid line) for clean Ni(111) 3p in the surface normal exit direction.

The Usage of MSCD Package

Source Code Description

We have developed a program package to calculate and analyze the photoelectron diffraction intensity and their chi function using C++ with Object Oriented Programming (OOP) and the parallel computing technique Message Passing Interface (MPI). The code features the multiple scattering approach developed by Rehr and Albers, the TPP-2 inelastic mean free path formula developed by Tanuma, Powell and Penn, and the correlated temperature effect developed by Sagurton, Bullock and Fadley. The code simulates various kinds of experimental scanning modes, sample or analyzer rotation mechanisms, chi calculation algorithms, and their combinations. The scanned-angle calculation and built-in fitting procedure fits most analysis needs of photoelectron diffraction data. By avoiding any repeat calculations for identical events and identical path summing, the program speeds up the calculation dramatically compared with several other codes.

To run the program on different platforms, we split our package into two parts: a machine independent source code; and a machine-dependent source code. The same machine independent source code applies to both sequential and parallel computers. Using this structure, we have maximized code portability. Because the machine-dependent source code is relatively quite small

(about 2 Kb), we can easily rewrite it to transfer the package to a new kind of computer.

The following diagram sketches the division of the code into a machine-independent part ("core program") and a machine-dependent part ("shell functions"). It gives examples of general calls in the core program that use machine-specific calls in the shell-functions part.

Parallel platforms	Sequential platforms
Cray T3E, COMPS	Cray J90, Sun workstation, IBM PC and Macintosh

Core Program (Machine Independent)

450 K source code, same for parallel or non-parallel

Use machine independent function calls everywhere

Example:

```
int main(int argc, char **argv)
{ mpiinit(argc,argv);
  begcputime=processortime();
  ... ... ...
  endcputime=processortime();
  mpiend();
  return(error);
}
```

Shell Functions (Machine Dependent)

2 K source code for each platform

CPU time usage, and basic MPI functions

Example for Cray T3E:

```
long processortime()
{ return(ta=times(&timebuf)); }

void mpiinit(int *argc,char ***argv)
{ MPI_Init(argc,argv); }

void mpiend()
{ MPI_Finalize(); }
```

Library functions, taken care of by C++ compiler
Example: Void MPI_Init(argc,argv);

Executable Programs

To perform scanned-energy calculation or fitting	
executable file	mscd
batch input file	mscdin.txt
include files	input files, phase shift files, radial matrix files, experimental scanned-energy data files (for fitting only)
output files	calculated scanned-energy data files

To calculate phase shift and radial matrix data files	
executable file	psrm
input file	psrmin.txt
include file	potential data file
output file	phase shift or radial matrix data files

To perform real space hologram transformation	
executable file	holo
input file	holoin.txt
include file	calculated or experimental scanned-energy data file
output file	real space image data

To calculate chi from experimental or calculated intensity data	
executable file	calchi
input file	scanned-energy data with intensity
output file	scanned-energy data with intensity and chi

To calculate normalized chi from experimental or calculated data file	
executable file	calnox
input file	scanned-energy data file
output file	scanned-energy data file

To calculate difference between two scanned-energy data	
executable file	caldif
input file	two scanned-energy data file
output on screen	reliability factors

To convert potential data to mscd format	
executable file	poconv
input file	simple format potential data (two columns)
output file	mscd format potential data

To convert phase shift data to mscd format	
executable file	psconv
input file	Fadley's group format phase shift data
output file	mscd format phase shift data

To convert radial matrix data to mscd format	
executable file	rmconv
input file	Fadley's group format radial matrix element data
output file	mscd format radial matrix element data

To calculate IMFP using TPP-2 formula or attenuation length	
executable file	calmfp
input on screen	follow instructions on screen
output file	mscd format mean free path data file

To calculate thermal vibrational mean square relative displacement	
executable file	calvib
input on screen	follow instructions on screen
output file	mscd format vibrational MSRD data file

To calculate photoelectron scattering factor	
executable file	calfac
input on screen	follow instructions on screen
output file	mscd scattering factor data file

Data Types and Unified Data Format

All the data files used or output by the mscd package programs are ASCII text file, having a unified mscd format data structure. This format consists of three parts.

1. The first part has only one line, i.e. the header line, including three integers, datatype, beginning-row, and line-numbers. Here, the three-digit datatype indicates what kind of data are in the file, beginning-row and line-numbers indicate the number of the first line counting from 1 for the head line and the number of total lines of the data body, i.e. the third part.
2. The second part, from the second line to the beginning-row of the data body, is a comment, introducing all the necessary information about the data.
3. The third part is the data body with a beginning line and line numbers indicated in the header line. The data are usually arranged in two or three columns, with spaces serving as the column delimiters.
4. We can put more comment lines after the third part.

Commonly used symbols and their definition in the mscd package:

symbol	**description**
k	photoemission wave vector in unit of $Å^{-1}$, k=0.512331*sqrt(kinetic energy in eV)
theta	photoemission detector polar angle in unit of degree
phi	photoemission detector azimuthal angle in unit of degree
I	angle-revolved photoemission intensity, in arbitrary unit
I_0	intensity background, in arbitrary unit
I_{0t}	intensity reference, i.e. the calculated photoemission intensity contributed by the emitters only without scatterers
I_{0b}	intensity background, i.e. a smooth function extracted from the calculated or experimental intensity data by using weighted sample spline fitting (for k or theta dependent curve), or a constant which is the average of the intensity data over all the azimuthal angles (for phi dependent curve only)
chi	non-dimensional function chi=$(I-I_0)/I_0$
l	angular momentum index
lmax	maximum value of angular momentum index
m	magnetic quantum number
l_i	initial angular momentum
l_f	final angular momentum ($l_f=l_i-1$ and l_i+1)
delta(l)	phase shift of a scattering event for angular momentum l
$R(l_f)$	amplitude of the radial component of the dipole matrix element into the given final state (l_f)
delta(l_f)	phase of the dipole matrix element into the given final state (l_f)

Definition of the three-digit datatype of scanned-energy data file:

	name	definition
third digit from left	1	energy dependent curves or hologram in (k I I_0 chi)
	2	polar angle dependent curves in (theta I I_0 chi)
	3	azimuthal angle dependent curves in (phi I I_0 chi)
	4	solid angle dependent hologram in (theta phi I I_0 chi)
	5-8	same as 1-4, but normalize chi
second digit from left	1	rotations of the geometry are made by rotating the analyzer in both polar and azimuthal directions
	2	rotations of the geometry are made by rotating the sample in both polar and azimuthal directions
	3-6	reserved for MCD implementation
first digit from left	1	calculated photoelectron diffraction pattern, chi is calculated by theoretical result chi=$(I-I_{0t})/I_{0t}$
	2	calculated photoelectron diffraction pattern, chi is calculated by theoretical result chi=$(I-I_{0b})/I_{0b}$
	3	experimental photoelectron diffraction pattern, chi is calculated by experimental data chi=$(I-I_{0b})/I_{0b}$
	4	same as 3, but no intensity I and background I_0 data
	5-6	reserved for MCD implementation

Date types of other files:

711	phase shift data in (k delta(l=0 1 2 ...))
721	radial matrix data in (k R(li+1) delta(li+1) R(li-1) delta(li-1))
731	calculation report
741	input data file for photoemission calculation
751	a batch of input files (filename should be mscdin.txt)
811	potential data in (r (angstrom) rV (eV-angstrom))
821	input file for phase shift or radial matrix (psrmin.txt)
831	subshell eigen wave function data in (r (angstrom) u (arb unit))
911	real space hologram image file in (x y z (angs) intensity (arb unit))
921	input file for hologram transformation (holoin.txt)
931	effective scattering factors as function of energy and scattering angle
941	electron inelastic mean free path data as function of energy
951	thermal vibrational mean square relative displacement data as function of bond length

General procedure to perform an scanned-energy calculation

1. Prepare the potential data for all kinds of investigated atoms, in two columns. The first column is the radial distance r in Bohr unit (1 Bohr = 0.529167 Å), the second column is the corresponding potential multiplied by the radial distance, rV(r), in Bohr-Rydberg unit (1 Rydberg = 13.605 eV, 1 Bohr-Rydberg = 7.19932 ÅeV).
2. Use application program poconv to convert the potential data prepared in step 1) to mscd format potential data file.
3. Prepare input file, psrmin.txt, for phase shift and radial matrix element calculation.

4. Use application program psrm to calculate phase shift and radial matrix data if needed.
5. Prepare input file for photoelectron diffraction pattern calculation.
6. Prepare batch file, mscdin.txt.
7. Use application program mscd to perform the photoelectron diffraction pattern calculation.
8. If needed, prepare input file, holoin.txt, for hologram transformation.

Preparing the input file for a scanned-energy calculation

Unit cell and cluster selection

The most complicated text file in mscd package is the input data file for a scanned-energy calculation. This file consists of non-structural parameters (like scanning mode, fitting mode, initial state, geometry, inelastic mean free path formula, temperatures, etc.) and structure parameters which determine the locations of atoms.

We define the surface structure by logical layers. A logical layer is a two-dimensional primitive lattice, or a two-dimensional Bravais lattice, which specifies the 2D-periodic array. A non-primitive physical layer must be divided into 2 or more primitive logical layers. Thus, a logical layer consists of all atoms with position vectors **R** of the form $\mathbf{R}(x,y,z) = m\ \mathbf{a}(x,y) + n\ \mathbf{b}(x,y) + \mathbf{R}_{\text{layer-origin}}(x,y,z)$, where **a** and **b** are two-dimensional vectors, m and n range through all integral values which can be negative, zero as well as positive. $\mathbf{R}_{\text{layer-origin}}(x,y,z)$ is the location of the sample origin. The choice of two-dimensional vectors is not unique.

If a physical layer consists of two kinds of atoms, the layer can be divided into two logical layers. Each logical layer can have only one kind of atom. If the atoms in a physical layer are emitters, basically all the atoms in that layer are emitters, by periodic equivalence. Since the photoelectron diffraction intensities from different emitters are independent, we only need to consider a few emitters with different neighboring scatterers. In our code of photoelectron diffraction calculation, we take only one emitter for each logical layer: it is the atom which sits at that layer's origin. This way, even a primitive emitter layer may have to be divided into several logical layers, if more than one emitters have to be considered because of different neighboring atoms in three dimensions. This is the case, for instance, when an overlayer has a 2D structure with a superlattice or disorder, making substrate atoms translationally inequivalent: then the emission from these different atoms must be calculated independently and summed.

The ideal structure of a surface is infinite in extent in two dimensions and infinitely deep. To correctly represent photoelectron diffraction, we must in principle therefore take account of all scatterers within the surface. In practice, the contribution from a scatterer far from the emitter may be ignored. So we only need to choose a finite cluster including a few emitters and a finite number of scatterers, typically 100 atoms. In this program package, we use a cluster shape defined by a semi-ellipsoid: the profile as seen parallel to the surface is a semi-ellipse of minor axis r and major axis h; perpendicular to the surface it has as cross-section a circle of radius r. All the atoms within this semi-ellipsoid will be taken into account in the calculation. All the atoms outside this cluster are ignored. A bigger cluster selection will give a more accurate calculation, because of the smaller edge effect. But a bigger cluster calculation will take much more computation time and require a much larger computer memory. We have to make a compromise between cluster size and computation time. Typically, a cluster of 80 to 100 atoms is a reasonable choice, which requires 10 MB to 16 MB of random memory for single precision floating point computation, or double that for double precision floating point processing.

For testing purposes, we have two other alternative choices for the cluster. We can draw different circles or rectangles for each logical layer, all the atoms on that layer within that circle or rectangle will be considered.

Non-structural parameters

The non-structural input parameters are initial core-level, number of multiple scattering orders, R-A approximation order, scanning mode (description of energy scanning, polar angle scanning, azimuthal angle scanning, or hologram, i.e. two-angle scanning, using the same codes for the data type), detector instrumental half-aperture angle, photon polarization angles, Debye temperature, density and atomic weight for each kind of atoms, and number of valence electrons for the inelastic mean free path calculation.
See detail of those definitions.

In the current version, we only allow linearly polarized photons (other light polarizations are linear combinations of two perpendicular linear polarizations, and can be produced as such). The direction of photon polarization is described by its polar and azimuthal angles, theta(p) and phi(p). They are defined as the polar and azimuthal angles with respect to a reference direction, which depends on the rotation mechanism of the experimental geometry. If the surface normal direction is fixed (for example do not rotate sample, or only rotate it in azimuthal direction), the reference direction is this fixed surface normal. If the surface normal is not fixed (rotatable), in a reasonable experimental geometry, the analyzer must be fixed, then the reference direction is this fixed analyzer direction.

Chi function and reliability factor

The photoemission intensity curves versus energy or angles consist typically of a series of diffraction peaks and troughs. To compare calculated with experimental intensities I, we have to remove the atomic partial cross section from intensities I_0, leaving only the oscillating part of the intensity, which we call chi function or χ function.

$$chi = (I-I_0)/I_0$$

Because the free atom cross-section I_0 contains only very low frequency information, we will make little error at the structurally important frequencies if we approximate I_0 as the smooth part of I.

More specifically, in our MSCD program package, to extract this background I_0 from an energy-scanned or polar-angle-scanned curve, we fit it as a cubic spline function to both calculated and experimental intensity curves using the non-linear Marquardt fitting method. To make this spline function, we choose 3-5 initial data points of equal step as fitting parameters, which ensure a smooth background and have only very low frequencies. For an azimuthal-angle-scanned curve, because the cross-section along azimuthal direction can be thought as a constant, we just use the average of the intensities as the background I_0.

Although the multiple scattering theory can well reproduce the experimental photoemission intensity, the agreement is never perfect. To quantify the goodness of fit or to automatically fit structural parameters to the data, we introduce the reliability or R-factor. In our MSCD package, we use two reliability factors Ra and Rb, defined as follows together with a similar factor R:

$$R = sum_j(chi_{cj} - chi_{ej})^2 / sum_j(chi_{ej})^2$$

$$R_a = sum_j(chi_{cj} - chi_{ej})^2 / sum_j(chi_{cj} + chi_{ej})^2$$

$$R_b = sum_j(chi_{cj}^2 - chi_{ej}^2) / sum_j(chi_{cj}^2 + chi_{ej}^2)$$

These R-factors are related to the traditional Pendry reliability factor for used in LEED; chi_c is the calculated chi value, while chi_e is the experimental one, the sum index j is taken over all the corresponding data points for both curves. The first factor R is used by many groups, Ra is the factor used in our fitting procedure, Rb is a factor which tells us the quality of amplitude agreement. The largest effect on the overall amplitude of the chi function stems from the thermal vibration: thus, from the Rb value, we will have an idea how good the Debye temperature is.

Built-in fitting procedure

In the complicated photoelectron diffraction theory, many parameters, such as temperature, inner potential, geometry and structural parameters, affect the photoemission intensity in a non-monotonous way. Thus, fitting may be the only effective method to perform the fitting of such parameters to experiment. A good reliability factor is then required to control the fitting process. In our MSCD package, we use the Ra factor, instead of R, to peform the fit, for the following reason.

For the best trial calculation, R and R_a are both small (<< 1.0). The amplitudes of calculated and experimental data are quite close, i.e.

$$R_{a\text{-}best} \sim R_{best} \qquad (\text{for Rbest} << 1.0)$$

For the worst trial calculation, the calculated intensity is irrelevant to the experimental curve. We would have

$$R_{worst} = 1 + A_{try}^2/A_{exp}^2$$
$$R_{a\text{-}worst} = 2.0$$

Here, Atry and Aexp are the average amplitude of the trial calculated and experimental intensity curves, respectively. Because Atry could be much less than Aexp in the fitting procedure while changing parameters, especially the Debye temperature, so we can write

$$R_{worst} = 1.0$$
$$R_{a\text{-}worst} = 2.0$$

Using the simplest single-scattering model of scanned-energy, chi can be written as

$$\text{chi}(k) = \text{sum}_j \{ A_j(k) \cos[k (R_j - R_j \cos(\text{theta}_j) + \text{phi}_j] \}$$

where k is the wave vector, A_j the amplitude, R_j the bond length, theta_j the scattering angle, and phi_j the phase of the jth scattering event. Let us consider only one component with same frequency k but with different amplitudes and phases for the trial and experimental curves, namely

$$\text{chi}_c(k) = A_{try} \cos(k x + \text{phi})$$
$$\text{chi}_e(k) = A_{exp} \cos (k x)$$

where x = R(1-cos(theta)), and chi_c and chi_e are calculated and experimental chi curves. From the R-factor definitions, we obtain

$$R / R_{worst} = [A_{try}^2 + A_{exp}^2 - 2 A_{try} A_{exp} \cos (\text{phi})] / A_{exp}^2$$
$$R_a / R_{worst} = [A_{try}^2 + A_{exp}^2 - 2 A_{try} A_{exp} \cos (\text{phi})] / [A_{try}^2 + A_{exp}^2]$$

Here we found $R_a/R_{a\text{-}worst} < R/R_{worst}$. We know, once a trial R-factor is greater than one of the extreme worst cases, that the fitting procedure would fail to continue towards the best minimum. The next figure shows an R-factor plot versus the fitting parameters. Assuming $A_{try} = A_{exp}$, delta(phi) in the figure will be the available fitting range of phase phi in the above equation. Obviously, we hope phi to be as large as possible. From the above equation, we find phi = 120° if we use R as R-factor and assume $A_{try} = A_{exp}$, and phi is even smaller when $A_{try} < A_{exp}$. However, we can see that phi = 180° if we use R_a as the R-factor to monitor the fitting process. This is the reason we use R_a as the monitoring R-factor in our fitting procedure.

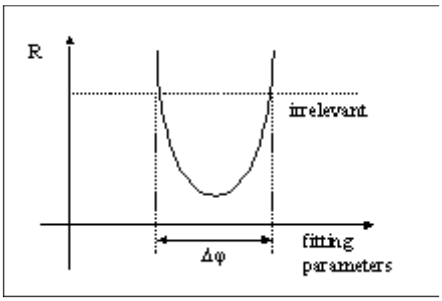


Figure. R-factor plot versus fitting parameters. If a trial R-factor is greater than the worst case (irrelevant), the fitting procedure would fail to continue.

For a poorly-known system, we may have several structural parameters to fit, such as layer spacing, location of adsorbate atom, surface reconstruction, etc. Because the inner potential and Debye temperature are not well understood so far, we have to keep them adjustable, too. All these parameters influence the photoemission intensity in a complicated way. Changing any of them may introduce more than one local minimum. Therefore, a simple conventional fitting method would not be enough in most cases. In our code, we perform the fitting in a three-step process.

1. In the first step, we do a net search, in which calculations are made on a set of given grid points in fixed stepsize basis for each fitting parameter, and their R_a factor is monitored. This allows us to search a larger area, even if there are more than one local minimum. Ending this process, the best minimum and its parameters are chosen as the initial parameters of the next step.
2. In the second step, the program uses a downhill simplex method in multi-dimensions monitoring the R_a factor. This method requires only function evaluations, not derivatives. It is not very efficient in terms of the number of function evaluations that it requires. However, it may frequently be the best method to use if the figure of merit (here R_a factor) is "get something working quickly" for a problem whose computational burden is small. Once a trial calculation gives an R_a factor less than a given (input) tolerance, the downhill simplex loop will stop and transmit the best parameters to the next step.
3. In the third step, we use the non-linear Marquardt fitting method, which is the most refined search in our package. This method works very well in practice and has become a standard non-linear least-squares approach.

MSCD Sample Data Files

1. Sample: Bohr-Rydberg format atomic potential data file

```
1.458753E-04    5.796773E+01
5.835011E-04    5.787076E+01
1.312877E-03    5.770851E+01
2.334004E-03    5.748013E+01
3.646882E-03    5.718506E+01
5.251508E-03    5.682329E+01
7.147886E-03    5.639558E+01
9.336017E-03    5.590395E+01
1.181589E-02    5.535141E+01
1.458753E-02    5.474217E+01
1.765091E-02    5.408101E+01
(more)
1.996886E+00    7.367796E-01
2.031167E+00    6.740353E-01
2.065740E+00    6.147618E-01
2.100603E+00    5.592108E-01
2.135759E+00    5.068808E-01
2.171207E+00    4.580389E-01
2.206946E+00    4.129712E-01
2.242978E+00    3.711888E-01
2.279301E+00    3.332452E-01
2.315915E+00    2.989336E-01
2.352821E+00    2.683100E-01
2.390020E+00    2.411752E-01
```

This is a typical data file of muffin-tin potential for a specific kind of atom. The first column is the radius in unit of Bohr radius (a_0=0.529167 Å), the second is the potential times the radius rV(r), in unit of Bohr-Rydberg (1 Rydberg energy = 13.605 eV). Calculated Electronic Properties of Metals (by V. L. Moruzzi, J. F. Janak, A. R. Williams, Pergamon Press, New York, 1978) listed calculated muffin-tin potentials for 32 elements which comprised of atoms possessing fewer than approximately 50 protons (H, Li, Be, Na, Mg, Al, K, Ca, Sc, Ti, V, Cr, Mn, Fe, Co, Ni, Cu, Zn, Ga, Rb, Sr, Yb, Zr, Nb, Mo, Tc, Ru, Rh, Pd, Ag, Cd, In). Barbieri and M.A. Van Hove developed a program that can calculate muffin-tin potential for all the elements in arbitrary given environment. Zabinsky, Rehr, Ankudinov and Albers have another code to do the same calculation .

This data format is not accepted by the MSCD package. One can use utility program poconv to convert it into acceptable MSCD format.

2. Sample: MSCD format atomic potential data file

```
811    12   128     datakind beginning-row linenumbers
-----------------------------------------------------------------
Conversion of traditional potential data to mscd format
POCONV Version 1.20 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
-----------------------------------------------------------------

        Cu              Copper potential data

    r (angstrom)   rV (eV-angs)
   7.719240e-05   4.173281e+02
   3.087696e-04   4.166300e+02
   6.947312e-04   4.154619e+02
   1.235078e-03   4.138177e+02
   1.929810e-03   4.116934e+02
   2.778925e-03   4.090889e+02
   3.782425e-03   4.060097e+02
   4.940312e-03   4.024702e+02
   6.252579e-03   3.984923e+02
   7.719240e-03   3.941062e+02
   9.340279e-03   3.893463e+02
   1.111570e-02   3.842518e+02
   1.304551e-02   3.788623e+02
   1.512970e-02   3.732152e+02
   1.736828e-02   3.673450e+02
   1.976125e-02   3.612812e+02
   2.230860e-02   3.550491e+02
   2.501033e-02   3.486708e+02
   2.786645e-02   3.421666e+02
   3.087695e-02   3.355566e+02
   3.404181e-02   3.288626e+02
   3.736110e-02   3.221061e+02
   4.083475e-02   3.153089e+02
   4.446280e-02   3.084899e+02
   4.824524e-02   3.016631e+02
 (more)
   1.056686e+00   5.304310e+00
   1.074827e+00   4.852594e+00
   1.093122e+00   4.425865e+00
   1.111570e+00   4.025936e+00
   1.130173e+00   3.649196e+00
   1.148931e+00   3.297567e+00
   1.167843e+00   2.973111e+00
   1.186910e+00   2.672306e+00
   1.206131e+00   2.399138e+00
   1.225506e+00   2.152118e+00
   1.245035e+00   1.931649e+00
   1.264720e+00   1.736297e+00
```

This is the MSCD format muffin-tin potential data file. The first column is the radius r, and second rV(r). The last radius data point is the muffin-tin radius. One can use utility program *psrm* to calculate the phase shift data and subshell radial matrix element data.

3. Sample: Atomic phase shift data file

```
711    13     0     datakind beginning-row linenumbers
-----------------------------------------------------------------
Conversion of traditional phase shift data file to mscd format
PSCONV Version 1.20 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
-----------------------------------------------------------------

Cu              Copper phase shift data

   parameters: number of wave vectors and quantum momenta
   columns: k (1/angstrom)   phase (l = 0 - 41) (radian)
      105       18
   3.6204  -1.1377e+000  -2.6770e-001  -1.8500e-001   1.8400e-001
            2.6100e-002   3.2000e-003   3.0000e-004   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000
   3.7971  -1.2115e+000  -3.1390e-001  -1.7090e-001   2.2800e-001
            3.5500e-002   4.9000e-003   5.0000e-004   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000
   3.9659  -1.2804e+000  -3.5900e-001  -1.5920e-001   2.7370e-001
            4.6600e-002   7.0000e-003   8.0000e-004   1.0000e-004
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000
   4.1279  -1.3452e+000  -4.0270e-001  -1.4980e-001   3.2000e-001
            5.9300e-002   9.7000e-003   1.3000e-003   1.0000e-004
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000
   4.2837  -1.4060e+000  -4.4500e-001  -1.4290e-001   3.6580e-001
            7.3300e-002   1.2900e-002   1.8000e-003   2.0000e-004
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000   0.0000e+000   0.0000e+000
            0.0000e+000   0.0000e+000
  (more)
  12.2238   2.8648e+000  -1.9691e+000  -4.4030e-001   1.5457e+000
            8.9490e-001   5.5110e-001   3.5510e-001   2.3660e-001
            1.5810e-001   1.0470e-001   6.9400e-002   4.4000e-002
            2.4600e-002   1.1600e-002   4.6000e-003   1.5000e-003
            4.0000e-004   1.0000e-004
```

This is a typical phase shift data file used in the *mscd* program. The lnum before the phase shift data body is the number of columns, i.e. number of angular momenta, of the phase shift data covered next. The first column is the wave vector in unit of $Å^{-1}$. Then there are lnum columns phase shift data for angular momentum l = 0, 1, 2, ... , lnum-1. This data file can be calculated from muffin-tin potential data by using utility program *psrm*.

4. Sample: Atomic radial matrix data file

```
721    14    49     datakind beginning-row linenumbers
---------------------------------------------------------------
MSCD Version 1.00 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
---------------------------------------------------------------

        Cu 3p           Copper radial matrix data

        binding energy =   60.000 eV

            k     R(li+1)    phase(li+1)     R(li-1)    phase(li-1)
       3.0000     .12621E-01      3.0354     .10288          2.3853
       3.2500     .50804E-02      3.0557     .97612E-01      2.2698
       3.5000     .21024E-01      3.0774     .92526E-01      2.1571
       3.7500     .34854E-01      3.0961     .87650E-01      2.0480
       4.0000     .46429E-01      3.1097     .82965E-01      1.9430
       4.2500     .55824E-01      3.1173     .78436E-01      1.8422
       4.5000     .63253E-01      3.1195     .74031E-01      1.7456
       4.7500     .68969E-01      3.1173     .69739E-01      1.6531
       5.0000     .73211E-01      3.1123     .65568E-01      1.5641
(more)
      13.0000     .30214E-01      2.7179     .11032E-01      -.3539
      13.2500     .28946E-01      2.7052     .10493E-01      -.3956
      13.5000     .27731E-01      2.6928     .99809E-02      -.4364
      13.7500     .26584E-01      2.6807     .95092E-02      -.4764
      14.0000     .25510E-01      2.6687     .90855E-02      -.5156
      14.2500     .24507E-01      2.6569     .87095E-02      -.5541
      14.5000     .23563E-01      2.6451     .83720E-02      -.5920
      14.7500     .22662E-01      2.6332     .80584E-02      -.6294
      15.0000     .21788E-01      2.6213     .77529E-02      -.6662
```

This is a typical radial matrix data file used in *mscd* program. The first column is wave vector in unit of $Å^{-1}$. The second and fourth columns are overlap of the radial components of the continuum orbital at $l_f=l_i\pm1$ and initial core orbital at quantum numbers (n_i,l_i). The third and fifth columns are phases of the dipole matrix element into the given final state $l_f=l_i\pm1$. This data file can be calculated from muffin-tin potential data by using utility program *psrm*.

5. Sample: single-curve experimental photoelectron diffraction data file

```
311    19    94     datakind beginning-row multi-curves
-----------------------------------------------------------------
MSCD Version 1.00 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
-----------------------------------------------------------------
 angle-resolved photoemission extended fine structure (ARPEFS)
 experimental data of Cu(111)-3p
 provided by Tony Huff (LBL) on March 10, 1995

   intial angular momentum (l) =  1
   photon polarization angle (polar,azimuth) = (   10.0,   0.0 ) (deg)
   sample temperature = 80 K

   photoemission energy scan curves
     (curve point theta phi weightc weighte//k intensity chiexp)
      1    94   94    1    1    1  ncurve npoint nk ntheta nphi nangle
    1    94    0.0    0.0   1.00    0.0   ----------------------------
     5.16    6.9078e+03    1.0875e+04   -3.6480e-01
     5.23    7.3998e+03    1.0743e+04   -3.1121e-01
     5.30    9.9001e+03    1.0611e+04   -6.7006e-02
     5.38    1.1573e+04    1.0460e+04    1.0641e-01
     5.45    8.9564e+03    1.0327e+04   -1.3273e-01
     5.52    8.5899e+03    1.0193e+04   -1.5731e-01
     5.59    1.3468e+04    1.0059e+04    3.3891e-01
     5.67    1.3365e+04    9.9038e+03    3.4948e-01
     5.74    9.6079e+03    9.7666e+03   -1.6247e-02
     5.81    1.0950e+04    9.6278e+03    1.3733e-01
     5.89    1.4849e+04    9.4671e+03    5.6849e-01
     5.96    1.2287e+04    9.3243e+03    3.1773e-01
     6.03    7.4261e+03    9.1795e+03   -1.9101e-01
     6.10    6.7878e+03    9.0324e+03   -2.4851e-01
     6.18    6.5566e+03    8.8613e+03   -2.6009e-01
     6.25    7.8330e+03    8.7089e+03   -1.0057e-01
     6.32    8.2807e+03    8.5537e+03   -3.1918e-02
     6.39    8.3385e+03    8.3957e+03   -6.8137e-03
(more)
    11.57    1.2998e+03    1.3109e+03   -8.4586e-03
    11.64    1.2859e+03    1.2659e+03    1.5817e-02
    11.71    1.1771e+03    1.2214e+03   -3.6271e-02
    11.79    1.1549e+03    1.1713e+03   -1.3993e-02
    11.86    1.1498e+03    1.1281e+03    1.9212e-02
    11.93    1.1678e+03    1.0856e+03    7.5674e-02
```

This is a typical single curve experimental energy scanning photoelectron diffraction data. The first data column is the wave vector in unit of $Å^{-1}$ for energy scanning, or degree for angle scanning, second the photoelectron intensity in arbitrary unit, and third the chi data. Comments are put between the top line and data body. This file is prepared by user. User only need to organize the first two columns above from experimental data, leaving third column blank or anything. Then use utility program calchi to make the final this file which will have three columns including the chi data.

6. Sample: multi-curve experimental photoelectron diffraction data file

```
321    16     0     datakind beginning-row linenumbers
-----------------------------------------------------------------
MSCD Version 1.00 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
-----------------------------------------------------------------
 angle-resolved photoemission extended fine structure (ARPEFS)
 experimental data of Au/Fe(001)-4f 7-1/2
 provided by Scot (UCB and LBNL) on January 2, 1996

   intial angular momentum (l) =  3
   photon polarization angle (polar,azimuth) = (   10.0,   0.0 ) (deg)
   sample temperature = 80 K

   photoemission energy scan curves
     (curve point theta phi weightc weighte//k intensity chiexp)
     2   129   66    2    1    2  ncurve npoint nk ntheta nphi nangle
   1    63    4.0    0.0    1.0    0.0   ----------------------------
     5.03    1.6123e+00    1.6515e+00   -2.3718e-02
     5.14    1.5889e+00    1.7767e+00   -1.0571e-01
     5.25    1.8433e+00    1.9047e+00   -3.2218e-02
     5.35    1.9856e+00    2.0235e+00   -1.8717e-02
     5.46    2.1073e+00    2.1570e+00   -2.3052e-02
(more)
    11.19    9.8578e+00    1.0033e+01   -1.7414e-02
    11.29    1.0529e+01    1.0203e+01    3.1927e-02
    11.39    9.9936e+00    1.0376e+01   -3.6839e-02
    11.49    1.0651e+01    1.0550e+01    9.5648e-03
   2    66   49.0   45.0    1.0    0.0   ----------------------------
     5.03    1.4651e+00    1.4996e+00   -2.3031e-02
     5.14    1.3372e+00    1.6738e+00   -2.0112e-01
     5.25    1.5963e+00    1.8570e+00   -1.4038e-01
     5.35    1.9787e+00    2.0312e+00   -2.5845e-02
     5.46    2.5285e+00    2.2313e+00    1.3320e-01
 (more)
    11.50    8.4234e+00    9.6168e+00   -1.2409e-01
    11.60    1.0726e+01    9.3203e+00    1.5082e-01
    11.71    1.0449e+01    8.9901e+00    1.6228e-01
    11.81    8.9943e+00    8.6881e+00    3.5249e-02
```

This is a typical multi-curve experimental photoelectron diffraction data. In the top line, linenumber = 0 means multi-curve format and there will be more detail information in the beginning line of data body. The beginning line includes the number of curves in this data file, number of total points, number of energies, polar angles, azimuthal angles, and solid angles. Here only the number of curves is actually important to the utility program, the other four numbers are information and their values do not matter with utility programs. This data file is prepared by user. User only need to organize the first two columns above from experimental data, leaving third column blank or anything. Then use utility program *calchi* to make the final this file which will have three columns including the chi data.

7. Sample: multi-curve calculated photoelectron diffraction data file

```
221    25     0     datakind beginning-row multi-curves
---------------------------------------------------------------
MSCD Version 1.00 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
---------------------------------------------------------------
 angle-resolved photoemission extended fine structure (ARPEFS)
 multiple scattering calculation of Au/Fe(001)-4f
 calculated by Yufeng Chen (LBNL) on Feb 27, 1996

   initial angular momentum (l) =  3   msorder=  8   raorder=  2
   photon polarization angle (polar,azimuth) = (   10.0,   0.0 ) (deg)

   radius, depth and lattice constant =   6.5,   8.1 and   2.87 angstrom
   cluster size =  75 atoms and spacings =   1.67   1.43   1.45 angstrom
   inner potential = 14.0 V  debye and sample temperature =  250 and   80 K
   number of valence electrons =   8     bandgap energy =   0.00 eV
   density of bulk =   7.86 g/cm3     molecular weight =   55.8 amu
   effective weight for kind 1-2 =   55.8  100.0
   half aperture angle =    0.0 deg            pathcut =   0.05

   photoemission energy scan curves
     (curve point theta phi weightc weighte//k intensity chical chiexp)
      2   130   65    2    1    2  ncurve npoint nk ntheta nphi nangle
    1    65    4.0    0.0   0.50    0.0   ----------------------------
    5.03   0.40234E-01   0.13601      -0.35894E-01
    5.13   0.36652E-01  -0.42057E-01  -0.11400
    5.23   0.37891E-01  -0.74581E-01  -0.53329E-01
    5.33   0.36154E-01  -0.17466      -0.22741E-01
    5.43   0.35347E-01  -0.24193      -0.29613E-01
    5.53   0.39516E-01  -0.19987      -0.12577E-01
    5.63   0.50471E-01  -0.30496E-01  -0.25516E-02
(more)
   10.63   0.22797E-01  -0.52359E-01  -0.44186E-01
   10.73   0.22457E-01  -0.26949E-01  -0.17402E-01
   10.83   0.22534E-01   0.19074E-01   0.10278
   10.93   0.21993E-01   0.39629E-01   0.24996E-01
   11.03   0.20697E-01   0.24205E-01  -0.58363E-01
   11.13   0.19270E-01  -0.37395E-04  -0.19668E-01
   11.23   0.18210E-01  -0.73002E-02   0.48760E-02
   11.33   0.17466E-01   0.65338E-02   0.68337E-02
   11.43   0.16877E-01   0.25399E-01  -0.36356E-01
    2    65   49.0   45.0   0.50    0.0   ----------------------------
    5.03   0.55098E-01  -0.34208E-01  -0.41377E-01
    5.13   0.45696E-01  -0.20223      -0.20562
    5.23   0.43294E-01  -0.28533      -0.16978
    5.33   0.48337E-01  -0.20216      -0.55992E-01
    5.43   0.63261E-01   0.43929E-01   0.84312E-01
    5.53   0.78572E-01   0.29598       0.23821
    5.63   0.85968E-01   0.41685       0.29789
 (more)
   10.63   0.22619E-01   0.46444E-01   0.70663E-01
   10.73   0.22961E-01   0.91457E-01   0.19225E-01
   10.83   0.22794E-01   0.11120       0.21271
   10.93   0.21638E-01   0.80177E-01   0.29176
   11.03   0.19488E-01  -0.50229E-02  -0.34999E-01
   11.13   0.17298E-01  -0.97507E-01  -0.21867
   11.23   0.16403E-01  -0.12589      -0.21111
   11.33   0.17629E-01  -0.58692E-01  -0.17506
   11.43   0.18816E-01   0.25788E-01  -0.93343E-01
```

```
fitted parameters ( nfit = 3 )

   rfac =  0.39062      afac =  0.18539      bfac =  0.50790E-01

    0    14.00    250.0    2.870           general vinner tdebye lattice
    1  0.58106  0.70711  1.00000  1.00000  layer spacing length unita unitb
    2  0.49878  0.00000  1.00000  1.00000  layer spacing length unita unitb
    3  0.50505  0.70711  1.00000  1.00000  layer spacing length unita unitb
    4  0.50000  0.00000  1.00000  1.00000  layer spacing length unita unitb
    5  0.50000  0.70711  1.00000  1.00000  layer spacing length unita unitb
    6  0.50000  0.00000  1.00000  1.00000  layer spacing length unita unitb
                                           unit: lattice constant

   rfac=sum((chic-chie)*(chic-chie))/sum(chie*chie)
   afac=sum((chic-chie)*(chic-chie))/(sum(chic*chic+chie*chie)
   bfac=sum(chic*chic-chie*chie)/sum(chic*chic+chie*chie)

 fitting history (159 trials )

   1  factors =     1.780993      0.9328943     -0.4761169E-01   Netsearch
      fitvars =    0.5235772      0.4477706      0.4550710
   2  factors =     1.651645      0.9087319     -0.1003962       Netsearch
      fitvars =    0.5235772      0.4477706      0.4803527
   3  factors =     1.714074      0.9636106     -0.1243512       Netsearch
      fitvars =    0.5235772      0.4477706      0.5056344
(more)
 127  factors =    0.3890323      0.1848499      0.4969384E-01   Downhill
      fitvars =    0.5817524      0.4975229      0.5056344
 128  factors =    0.7254332      0.3553312      0.2036142E-01   Downhill
      fitvars =    0.5526648      0.5223990      0.5309161
 129  factors =    0.7458019      0.3875329     -0.3923824E-01   Downhill
      fitvars =    0.6108400      0.4726467      0.5574619
(more)
 156. factors =    0.3650445      0.1877514     -0.2864927E-01   Marquardt
      fitvars =    0.5895040      0.4985123      0.5076532
 157. factors =    0.3410145      0.1778389     -0.4299886E-01   Marquardt
      fitvars =    0.5826772      0.5044975      0.5076532
 158. factors =    0.3933392      0.1863439      0.5250285E-01   Marquardt
      fitvars =    0.5810603      0.4987825      0.5111004
 159. factors =    0.3906236      0.1853918      0.5079023E-01   Marquardt
      fitvars =    0.5810603      0.4987825      0.5050498

This calculation took 007 CPU hours on a Sun Unix system
     starting on Mon Feb 26 09:16:28 1996
   and ending on Tue Feb 27 06:55:46 1996
```

This is a typical calculation output of photoelectron diffraction data after a fitting procedure using mscd program. The first part is the top line, header. The linenumber=0 means multi-curve format and there will be more information in the beginning line of data body. The second part is the comment before the data body, which includes all the necessary parameters used in the calculation. The third part is the photoelectron diffraction data, with the first column wave vector in unit of $Å^{-1}$ for energy scanning (degree for angle scanning), second column intensity in arbitrary unit, third column the chi value, and fourth the corresponding experimental chi value. Following is the fourth part, detail information of the fitting procedure, includes the best fitted parameter and R-factors, the history of the fitting procedure (their try values and R-factors), and the CPU time, starting, ending time, machine name carried out the calculation.

8. Sample: Batch input data file for MSCD calculation

```
751    10    24     datakind begining-row linenumbers
-----------------------------------------------------------------
MSCD Version 1.20 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
-----------------------------------------------------------------

     Batch of input files for calculation

in01    inytb001.txt    input data file
in02    inytb002.txt    input data file
in03    inytb003.txt    input data file
in04    inytb004.txt    input data file
in05    inytb005.txt    input data file
in06    inytb006.txt    input data file
in07    inytb007.txt    input data file
in08    inytb008.txt    input data file
*n09    inytb009.txt    input data file
*n10    inytb010.txt    input data file
*n11    inytb011.txt    input data file
*n12    inytb012.txt    input data file
*n13    inytb013.txt    input data file
*n14    inytb014.txt    input data file
*n15    inytb015.txt    input data file
*n16    inytb016.txt    input data file
in17    inytb017.txt    input data file
in18    inytb018.txt    input data file
in19    inytb019.txt    input data file
in20    inytb020.txt    input data file
*n21    inytb021.txt    input data file
*n22    inytb022.txt    input data file
*n23    inytb023.txt    input data file
*n24    inytb024.txt    input data file
in25    inytb025.txt    input data file
in26    inytb026.txt    input data file
in27    inytb027.txt    input data file
in28    inytb028.txt    input data file
in29    inytb029.txt    input data file
in30    inytb030.txt    input data file
*n31    inytb031.txt    input data file
*n32    inytb032.txt    input data file
```

This is a typical batch file including a batch of jobs for *mscd* program. The first column is the introductory name telling mscd program using the following data file (second column) as input data file for this job. If the introductory name start with *, the job will be skipped.

9. Sample: Input file for phase shift and radial matrix element calculation

```
821    11    11     datakind beginning-row linenumbers
 ---------------------------------------------------------------
                 David A. Shirley's group
              Pennsylvania State University (PSU)
          Lawrence Berkeley National Laboratory (LBNL)
     Copyright (c) 1995-1996 DAS group. All rights reserved
 ---------------------------------------------------------------

     input file for phase shift or radial matrix calculation

'po'    'pofe.txt'              input potential data file
'ps'    'psfe.txt'              output phase shift data file
'rm'    'rmfe3p.txt'            output radial matrix data file
'ei'    'eife3p.txt'            output eigen function data file
'ss'    '3p'                    subshell and initial state
'sb'    'Fe'                    symbol of atom
'at'    'Iron'                  name of atom

20      1                       lnum,outputfile (0 phase 1 radial matrix)
3.0     15.0    0.25            kmin,kmax,kstep
100                             subshell binding energy
```

This is a typical input data file for utility psrm program. Here lnum is the number of total angular momentum components for phase shift calculation. The outputfile=0 means to calculate phase shift data, outputfile=1 to calculate radial matrix element for the given subshell. The kmin, kmax and kstep are minimum, maximum and step of the wave vectors in unit of $Å^{-1}$ for the phase shift data or radial matrix element data. The subshell binding energy is a initial trial binding energy which will be fitted by the program automatically. The final binding energy fitted by the program will listed in the output radial matrix element data file. This theoretical binding energy is not necessary to equal the experimental one, but will not be far away. User should check it out see if it is reasonable. If the input binding energy are far away from its real value, the program will stop and display a message saying that *binding energy too large, program terminated* or *binding energy too small, program terminated*. In these cases, user should change the input binding energy to meet the program need. In some cases, if the binding energy is extremely small (like <1 eV), the program may fail to calculate the subshell radial matrix element.

10. Sample: Input data file for hologram transformation

```
921    11     8     datakind beginning-row linenumbers
 ---------------------------------------------------------------
                 David A. Shirley's group
              Pennsylvania State University (PSU)
          Lawrence Berkeley National Laboratory (LBNL)
     Copyright (c) 1995-1996 DAS group. All rights reserved
 ---------------------------------------------------------------

     input file for real space hologram transformation

'sn'    'Mn-O-Mn(100)'  system name
'pe'    'exmnosh1.txt'          input photoemission chi data file
'ho'    'homnosh1.txt'          output real space hologram file

10.0    25.0    0.25            vinner(eV) cone-angle (deg) k-window
-5.0    5.0     0.2             xmin xmax xstep (angstrom)
0.0     0.0     0.0             ymin ymax ystep (angstrom)
-5.0    0.0     0.2             zmin zmax zstep (angstrom)
```

This is a typical input data file for hologram inversion utility program holo. The vinner is the inner

potential in unit of eV. The cone-angle is full angle of the small window or small cone cross section in unit of degree. The cone-angle usually takes 25°-60°. The k-window is a Hanning window function parameter, which takes a value between 0.0 and 1.0. The k-window=0.0 means no window factor at all. The recommended value is 0.25. The xmin, xmax, xstep are minimum, maximum and step size of the points in x-direction where the real space hologram intensity will be calculated. The ymin, ymax, ystep, and zmin, zmax, zstep have the same definitions in y and z directions to define a real space of calculation. This small window method takes a set of energy scanning spectra on a grid over the full-emission hemisphere, providing a high-quality real space atomic image comparied with the constant-initial-energy spectra method. The numerical position values could be off by about 0.2 Å. If better numbers are desired, trial and error modeling (fitting) of the spectra could be done.

11. Sample: Input data file for MSCD calculation

```
741    10    92     datakind begining-row linenumbers
-------------------------------------------------------------
MSCD Version 1.00 Yufeng Chen and Michel A Van Hove
Lawrence Berkeley National Laboratory (LBNL), Berkeley, CA 94720
Copyright (c) Van Hove Group 1997. All rights reserved
-------------------------------------------------------------

     Au/Fe(001)-4f              Gold/Iron input file

un    "Yufeng Chen (LBNL)"      user name
sn      Au/Fe(001)-4f           system name
ps01    psfe.txt                input phase shift data file
ps02    psau.txt                input phase shift data file
rm      rmau4f.txt              input radial matrix data file
ex      exaufefk.txt            input experimental data file
pe      peaufefk.txt            output photo emission data file

221     0       0.1             scanmode,dispmode,ftolerance

3       0       8       2       linitial,lnum,msorder,raorder
9       0       0       0       layers,finals,fitmath,trymax
5.0     12.0    0.1             kmin,kmax,kstep (per angstrom)
0.0     45.0    45.0            dthetamin,dthetamax,dthetastep (degree)
45.0    0.0     0.0             dphimin,dphimax,dphistep (degree)
10.0    0.0     1               ltheta, lphi, beampol (degree)
0.0     0.0     0.0             mtheta, mphi, acceptang (degree)
6.5     0.0     2.87            radius,depth,lattice(angs)
8       0.0     7.86    55.8    valence,bandgap(eV),density(g/cm3),mweight
55.8    197.0   55.8    55.8    effective weight for kind 1-4 (amu)
0.0     0.0      0.0     0.0    magnetization amplitude for kind 1-4
14.0    250.0   80.0    0.01    vinner(eV),tdebye,tsample(K),pathcut
0.0     0.0     0.0             fit try for vinner, tdebye and lattice

1       2       1       0       layer, kind, emitter, lineatom
0       0       0       0       latoms(xa,xb,ya,yb)
 1.0000000        0.000000      unita(len ang) (bcc (001) structure)
 1.0000000       90.000000      unitb(len ang) (in unit of lattice)
 0.7071068       45.000000      origin(len ang) (in unit of lattice)
 0.0000000                      interlayer spacing (unit lattice)
 0.0    0.0      0.0            fit try for spacing, length and units

2       1       0       0       layer, kind, emitter, lineatom
0       0       0       0       latoms(xa,xb,ya,yb)
 1.0000000        0.000000      unita(len ang) (bcc (001) structure)
 1.0000000       90.000000      unitb(len ang) (in unit of lattice)
 0.0000000       0.0000000      origin(len ang) (in unit of lattice)
 0.5000000                      interlayer spacing (unit lattice)
```

```
 0.0    0.0      0.0            fit try for spacing, length and units

3       1       0       0      layer, kind, emitter, lineatom
0       0       0       0      latoms(xa,xb,ya,yb)
 1.0000000        0.000000      unita(len ang) (bcc (001) structure)
 1.0000000       90.000000      unitb(len ang) (in unit of lattice)
 0.7071068       45.000000      origin(len ang) (in unit of lattice)
 0.5000000                      interlayer spacing (unit lattice)
 0.0    0.0      0.0            fit try for spacing, length and units

4       1       0       0      layer, kind, emitter, lineatom
0       0       0       0      latoms(xa,xb,ya,yb)
 1.0000000        0.000000      unita(len ang) (bcc (001) structure)
 1.0000000       90.000000      unitb(len ang) (in unit of lattice)
 0.0000000       0.0000000      origin(len ang) (in unit of lattice)
 0.5000000                      interlayer spacing (unit lattice)
 0.0    0.0      0.0            fit try for spacing, length and units

 (more)
```

This is a typical input data file for mscd program. This file is prepared by user. Here is a list of input parameters and their definitions.

scanmode	Same as data_type, describes chi calculation algorithm, rotation mechanism, and scanning direction.
dispmode	This parameter is used to control the program displaying intermediate messages while in calculation. In a real calculation, user can set it to 0, which only display the necessary information on screen. Setting it to 1 will disable any information display, which is used for some supercomputer. Setting it to 8 will display all the information the program can. Setting it's ten's digit (for example 10) will have the program write all intermediate infromation into a text file *mscdlist.txt*.
ftolerance	This is tolerance parameter for controlling the Simplex Downhill and Marquardt fitting processes. Setting to 0.0 means no fitting. When the R-factor between calculation and experimental chi curves is less than sqrt(ftolerance), the Simplex Downhill process stops and fitting switches into Marquardt process. When R-factor is less than ftolerance, fitting process ceases.
linitial	This is the initial state l_i of the photon excitation process. linitial=0 for s core level, 1 for p level, 2 for d level, and 3 for f level.
lnum	This number is the number of angular momenta the mscd program will take into account in the scattering event calculation. For a real calculation, the lnum must be greater than $k*r_{mt}$, where k is wave vector and r_{mt} muffin-tin radius, which is roughly half of the inter-atomic length. Typically user need to set it to 20. Setting it to 0 means counting as many as available in the phase shift data file.
msorder	This is the multiple scattering order the program will take into account. Setting it to 0 means only calculate the reference wave, ignoring all the scatterers. Setting it to 1 means single scattering. Setting it to 2 means double scattering. For a real calculation, msorder is typically set to greater than 5. Setting it to 8 is recommended here.
raorder	This is the Rehr-Albert approximation order to describe the curved-wave expansion. Setting it to 2 is recommended in a real calculation. Setting it to 0 equivalent to point scattering theory. Setting it to -1 equivalent to plane wave theory.

layers	This is the number of total logical layers of the system.
finals	This parameter is for test only. User should use 0 to perform a real calculation. Setting it to 1 means accounting only (l_i+1) final state only, 2 accounting (l_i-1) state only, 3 accounting reference only, ignoring all the scatterers, 4 accounting only scattering wave, ignoring reference wave.
fitmath	This is a parameter to choose the method of fitting process. Setting it to 1 means to do finest Marquardt fitting only, 2 to do both fine Simplex downhill fitting and then the finest Marquardt fitting, 3 to do all the coarse grid search and then the fine Simplex downhill and then the finest Marquardt fittings, 4 to do coarse net search only, 5 to do a net search of emission angle deviation. User can set it to 0 means default process, which is equivalent to 2 in the current version.
trymax	This is the maximum number of tries of the fitting procedure. If the R-factor never be less than the given tolerances, the fitting procedure will cease after doing a number of tries defined by trymax. Setting it to 0 means default, which value depends on the fitting method and number of fitting parameters.
(kmin, kmax, kstep), (dtmin, dtmax, dtstep), (dpmin, dpmax, dpstep)	These are minimum, maximum and stepsize of wave vector k in unit of $Å^{-1}$, and theta, phi angles of the analyzer in unit of degree. The minimum value must be less than the maximum value. If the stepsize=0.0, the maximum value will adjusted automatically to equal the minimum value.
ltheta, lphi, mtheta, mphi	The polar and azimuthal angles of the photon polarization (e vector) and magnetization (theta(p) and phi(p), theta(m) and phi(m)) with respect to a reference direction in unit of degree. The surface normal serves this reference direction if the surface normal is fixed, otherwise the analyzer direction serves this reference. The mtheta and mphi have not implemented in this version. Set them to 0.0.
beampol	The polarization of the photon beam source. Set beampol to 0 or 1 means linear polarization, 2 means left circular, 3 means right circular, 4 for both left and right circular.
accepang	This is the half angle of the cross section of the analyzer instrumental aperture in unit of degree.
radius, depth	These are radii of both axes of the semi-ellipse section of the cluster in unit of Å. On the first layer, the cluster covers a circle with this radius. Setting depth=0.0 means using default value which depends the analyzer direction. Changing this radius will change the cluster size. When doing a batch of calculations on same sample but different emission angles, you had better to set a fixed radius and a fixed depth values to ensure the clusters are same for all these calculations. For a real calculation, cluster size should be at least 70 atoms.
lattice	This is the lattice constant of the crystal structure in unit of Å, serving the unit for all other length structural parameters. Because it serves unit for others, it is not necessary to use the same definition as in crystallography.
valence	This is the number of valence electrons of the molecular or atom in bulk when using TPP-2 formula, or the exponent of energy m (0.5-0.9) when using attenuation equation lamda=k*E^m. Setting it to 0 means ignoring inelastic scattering effect.
bandgap	This is the band gap energy in eV (equals 0.0 for metal) when using TPP-2 formula, or the coefficient of energy k (0.02-0.3) using attenuation equation

	lamda=k*E^m.
density	This is the density of the bulk material in (g/cm3).
mweight	This is the molecular weight of the bulk material (in amu).
effective weight	This is the effective atomic weight for specified atom (in amu).
magnetization amplitude	This is the magnetization amplitude for specified atom. This feature is not available in this version, set them to 0.
vinner	This is the inner potential in eV.
tdebye and tsample	These are debye and sample temperatures of the bulk in K.
pathcut	Pathcut is a threshold to cut those small contribution events or paths. Set it to zero means doing full calculation without any cut. A non-zero value means that the higher orders of multiple-scattering would not be calculated. Generally, lower kinetic energies will require lower pathcut values to obtain the full multiple-scattering calculation results. For simulating typical energy-scan photoelectron diffraction data where k > 5 $Å^{-1}$ (kinetic energy (KE) > 100 eV), pathcut = 0.05 works well if you are most interested in optimizing the calculation speed. However, note that for k < 4 $Å^{-1}$ (KE < 60 eV), pathcut = 0.01 may be too high for some applications. For low-energy angle-scan photoelectron data where 3 $Å^{-1}$ < k < 4 $Å^{-1}$ (40 eV < KE < 60 eV), you may need to set the pathcut as low as 0.001 to obtain the correct multiple-scattering calculation results. The general recommendation is to set pathcut = 0.01. But be alert! The world needs more lerts.
fit try for vinner, tdebye and lattice	These parameters are used to choose the actual fitting parameters. Setting to 0.0 means no fit for the corresponding parameter. A positive value serving the relative fitting stepsize for net search or simplex downhill fitting process. The stepsize equals this value times the corresponding initial value. The negative value will be treated as 0.0.
layer, kind, emitter, lineatom, latoms(xa,xb,ya,yb)	Here layer indicates the serial number of the current logical layer. The kind indicates the kind of atom of current layer (kind=1 always refer to bulk atom). The emitter=1 indicates that there is emitter in this logical layer, emitter=0 means no emitter. The lineatom and latoms(xa,xb,ya,yb) are typically for test only. When cluster radius set to 0.0, they provide two alternative way to choose the cluster for calculation.
unita, unitb (length and angle)	These parameters define the unit cell vectors for the current logical layer, consists of their length in unit of lattice constant and the angle with respect to the x-axis direction.
origin (length and angle)	The length and angle determine the origin of the current logical layer with respect to the absolute sample origin. The length uses unit of lattice constant, angle uses degree.
interlayer spacing	This is the spacing between current and previous logical layer, in unit of lattice constant. The layer spacing of the first logical layer is the spacing between the first layer and the virtual surface plane.
fit try for spacing, length and units	Here are more fitting parameters to control fitting process for inter-layer spacing, length between the layer origin and sample origin, and scaling factor for both unit vectors. Setting to 0.0 means no fit. A positive value serving the relative fitting stepsize for net search or simplex downhill fitting process. The stepsize equals this value times the corresponding initial value. The negative value means the corresponding parameter of current logical layer always equal the corresponding parameter of the previous logical layer.

Questions and Answers regarding the MSCD package

---

1. Is the polar emission angle defined with respect to the normal to the surface (i.e. θ=0 means normal emission)?
   Yes.
2. The ltheta, lphi, beampol refer to the light polarization. How can one treat the case of unpolarized x-rays (as in the case of a conventional x-ray source)?
   The latest version (Version 1.37) cannot handle unpolarized x-rays at this moment. We will have a version to cover this feature later. However, an unpolarized x-ray beam can be mathematically composed of two mutually perpendicular polarized light beams. So you can calculate separately for those two polarizations and finally add the intensities together.
3. What are mtheta, mphi and acceptang?
   The mtheta and mphi have not be implemented. Currently they should be set to zero. The acceptang is the half aperture angle of the photoemission energy detector. We usually choose 2 to 5 degrees.
4. How to set up the cluster radius and depth?
   These two parameters determine the shape of the cluster. If you set the depth to zero, then the program will determine its non-zero value. You have to set a non-zero radius. You can try different values (in unit of Angstrom), until the cluster size is between 80 to 120 atoms (a good compromise between calculation time and accuracy). The MSCD program shows the cluster size and some other parameters before the calculation.
5. What are the lineatom and latoms in the input file?
   The lineatom and latoms are reserved for test purposes only. The user can set them to zero.
6. How to build a layer in the input file, when the layer consists of more than one kind of atom or when the unit cell has more than one kind of atom?
   We build up a structure as a set of logical layers. Each logical layer has only one atom per unit cell (it thus forms a Bravais lattice). One physical layer consists of one or more logical layers, which may have interlayer spacings equal to zero, i.e. may be coplanar.
7. How to calculate phase shift data and radial matrix elements?
   There are several programs that can calculate phase shifts and radial matrix elements. We have a program, called psrm, which is able to calculate these data based on known muffin tin potentials. The psrm program, written in Fortran 77, is not included in the downloadable package. Users can request it for free by sending email to us.
8. What are kmin, kmax and kstep?
   The kmin, kmax, kstep are minimum, maximum and step of the photoelectron wave numbers. The relationship between wave number k, photoelectron kinetic energy E_k, photon energy E_p, and the core level binding energy E_b is
   $E_k = E_p - E_b$
   $k = 0.51233 \sqrt{E_k}$
   Here we use the unit eV for energies E_k, E_p and E_b, and $Å^{-1}$ for k. Typically at LBNL, the photoelectron diffraction scanned energy curves have kmin around 3 - 5 $Å^{-1}$, kmax around 10 - 15 $Å^{-1}$, and kstep around 0.1 or 0.2 $Å^{-1}$.
9. What compiler should we use to compile the MSCD source files?
   The source code is written in Ansi C++. We believe most of the popular C++ compilers can be used to compile this package. We have used the PC Microsoft Visual C++ compiler, the PC Borland C++ compiler, the Unix C++ compiler, the Macintosh CodeWarrier C++

compiler, and the Macintosh Symantec C++ compiler. The version of those compilers should be not important.

10. I have tried to compile the program with the MS Visual C++ (Version 2.2), but in this version there is no file strstream.h. Where can I get this header file?
In MS Visual C++, the file strstream.h is replaced by strstrea.h. Therefore, the only thing you need to do is to make a copy of strstrea.h to strstream.h in the same directory.

11. How much computer memory does the program need?
The computer memory required depends on the structure, cluster size, multiple scattering order, R-A approximation order and *pathcut*. A larger problem size needs more memory. Typically, for a full theoretical calculation of a 100-atom cluster with 2nd R-A approximation order for up to 8th multiple scattering order, about 32 megabytes of memory will be needed for one processor and *pathcut* = 0.01.

MSCD Package Installation and Release Notes

Supported Platforms and Requirements

- **Parallel platforms**
  Cray T3E supercomputer running Unix
  COMPS network of workstations running Unix
- **Sequential platforms**
  Cray J90 supercomputer running Unix
  Sun workstations running Unix
  PC Windows DOS system
  Macintosh Power PC
- **Other platforms**
  On other platform, you may write your own userxxxx.cpp file. Instructions for creating your own userxxxx.cpp file are provided here.
- **Disk space requirement**
  The final binaries of the package require 3 to 10 megabytes of hard disk space, depending on platform and compiling options.
- **Memory space requirement**
  The memory requirement depends on the cluster size, multiple scattering order and Rehr-Albers approximation order. To perform an accurate calculation (cluster size around 100, multiple scattering order 8, Rehr-Albers approximation order 4, pathcut 0.01), about 20 megabytes of memory space are needed.
- **Calculation time**
  The calculation time is very dependent on machine and problem size.

Installation

Download software

- We offer free download of our MSCD package, containing the source code.

Unzip downloaded package

- **To unzip the downloaded package on a PC Windows system:**
  winzip
- **To unzip the downloaded package on a Unix system:**
  unzip
- **To unzip the downloaded package on a Macintosh OS system:**
  unzip
- **More information about unzip can be found elsewhere on the internet**

Compile source code

- **To compile the source code on a PC under Windows DOS system:**
  make -f mscdpc.mak
- **To compile the source code on a Sun Workstation under Unix:**
  make -f mscdsun.mak
- **To compile the source code on a Cray J90 sequential supercomputer under Unix:**
  make -f mscdj90.mak
- **To compile the source code on a Cray T3E parallel supercomputer under Unix:**

make -f mscdt3e.mak

- **To compile the source code on a COMPS network of workstations under Unix in parallel mode:**
  make -f mscdcomp.mak
- **To compile the source code on another platform:**
  Typically, you need to create a project which includes program names and their needed source files. Here is the list:
  - **calchi**
    userinfo.cpp userutil.cpp cartesia.cpp polation.cpp curvefit.cpp pdinten.cpp pdintena.cpp calchi.cpp
  - **calnox**
    userinfo.cpp userutil.cpp cartesia.cpp polation.cpp curvefit.cpp pdinten.cpp pdintena.cpp calnox.cpp
  - **calnox**
    userinfo.cpp userutil.cpp cartesia.cpp polation.cpp curvefit.cpp pdinten.cpp pdintena.cpp calnox.cpp
  - **caldif**
    userinfo.cpp userutil.cpp cartesia.cpp polation.cpp curvefit.cpp pdinten.cpp pdintena.cpp caldif.cpp
  - **calfac**
    userinfo.cpp userutil.cpp polation.cpp phase.cpp msfuncs.cpp rotamat.cpp fcomplex.cpp scatter.cpp calfac.cpp
  - **poconv**
    userinfo.cpp userutil.cpp potentia.cpp poconv.cpp
  - **psconv**
    userinfo.cpp userutil.cpp fcomplex.cpp phase.cpp psconv.cpp
  - **rmconv**
    userinfo.cpp userutil.cpp radmat.cpp rmconv.cpp
  - **calmfp**
    userinfo.cpp userutil.cpp meanpath.cpp calmfp.cpp
  - **calvib**
    userinfo.cpp userutil.cpp vibrate.cpp calvib.cpp
  - **mscd**
    userinfo.cpp userutil.cpp cartesia.cpp polation.cpp curvefit.cpp pdinten.cpp pdintena.cpp fcomplex.cpp msfuncs.cpp pdchifit.cpp vibrate.cpp meanpath.cpp phase.cpp radmat.cpp rotamat.cpp mscdrun.cpp mscdruna.cpp mscdrunb.cpp mscdrunc.cpp mscdrund.cpp mscdrune.cpp mscdjob.cpp mscdmain.cpp jobtime.cpp
    userpc.cpp (for PC only)
    usersun.cpp (for Sun workstation only)
    usermac.cpp (for Macintosh only)
    userj90.cpp (for Cray J90 only)
    usert3e.cpp (for Cray T3E only)
    usercomp.cpp (for COMPS network of workstations only)

Develop Source Code for Other Platforms

- For each platform, there is only one userxxxx.cpp file, which is platform dependent, e.g. userpc.cpp for a PC, usersun.cpp for a Sun Unix system, etc. If you have a platform that is not listed above, you may develop your own userxxxx.cpp file. You will need to implement the following 9 functions that are defined in the userutil.h header file:
- 27. int computer(char *name=NULL,int size=0) --- copy name of computer into name with length limit of size, and return code of the computer.
  As minimum coding, you may copy this function from userpc.cpp and replace the computer

name "an IBM Personal Computer" with your actual computer name.

- 28. long timeprocessor() --- return current CPU time in seconds
  As minimum coding, you may make an empty code in this function and return zero for CPU time.
- 29. int mpiinit(int *argc,char ***argv) --- initialization of message passing interface for parallelization
  As minimum coding, you may copy this function from userpc.cpp without any change.
- 30. int mpigetmype() --- get id or rank of the current processor element
  As minimum coding, you may copy this function from userpc.cpp without any change.
- 31. int mpigetnumpe() --- get total number of processor elements
  As minimum coding, you may copy this function from userpc.cpp without any change.
- 32. int mpisend(char *buffer,int bufsize,int destpe,int tag) --- send out a memory buffer with size bufsize to the destination processor destpe with message tag (tag)
  As minimum coding, you may copy this function from userpc.cpp without any change.
- 33. int mpireceive(char *buffer,int bufsize,int sourcepe,int tag) --- receive from source processor element sourcepe a memory buffer with size bufsize and put in buffer
  As minimum coding, you may copy this function from userpc.cpp without any change.
- 34. int mpiend() --- finalize the message passing interface
  As minimum coding, you may copy this function from userpc.cpp without any change.
- 35. int sendemail(char *mailpath,char *message,char *stamp, char *fromaddress=NULL,char *toaddress=NULL,char *subject=NULL, char *ccaddress=NULL,char *bccaddress=NULL,int stamp=1) --- send email from fromaddress to toaddress, cc to ccaddress, bcc to bccaddres, with subject, message and stamp if stamp != 0
  As minimum coding, you may copy this function from userpc.cpp without any change.

Release Notes

- This program package was written by Yufeng Chen and Michel A. Van Hove at Lawrence Berkeley National Laboratory. We offer free download. You can freely copy, change and use it. However, the program package or its components may not be sold, leased, licensed or otherwise traded or commercialized in any way.
- Please acknowledge use of this package, called MSCD photoelectron diffraction program package, as: "Yufeng Chen and Michel A Van Hove, private communication".
  A useful reference is: Y. Chen, F.J. García de Abajo, A. Chassé, R.X. Ynzunza, A.P. Kaduwela, M.A. Van Hove and C.S. Fadley, Phys. Rev. B 58, 13121 (1998).
- The authors do not guarantee that the calculation results obtained from this program package will be correct.

Known problems or bugs

- **None**