

The effect of self-consistent potentials on EXAFS analysis

B. Ravel,^{a*} M. Newville,^b J. J. Kas^c and J. J. Rehr^c

^aNational Institute of Standards and Technology, Gaithersburg, MD 20899, USA, ^bCenter for Advanced Radiation Studies, University of Chicago, Chicago, IL 60637, USA, and ^cDepartment of Physics, University of Washington, Seattle, WA 98195, USA. *Correspondence e-mail: bravel@bnl.gov

Received 19 April 2017

Accepted 8 August 2017

Edited by S. M. Heald, Argonne National Laboratory, USA

Keywords: EXAFS; real-space multiple scattering; data analysis.

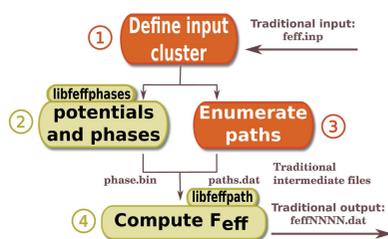
Supporting information: this article has supporting information at journals.iucr.org/s

A theory program intended for use with extended X-ray-absorption fine structure (EXAFS) spectroscopy and based on the popular *FEFF8* is presented. It provides an application programming interface designed to make it easy to integrate high-quality theory into EXAFS analysis software. This new code is then used to examine the impact of self-consistent scattering potentials on EXAFS data analysis by methodical testing of theoretical fitting standards against a curated suite of measured EXAFS data. For each data set, the results of a fit are compared using a well characterized structural model and theoretical fitting standards computed both with and without self-consistent potentials. It is demonstrated that the use of self-consistent potentials has scant impact on the results of the EXAFS analysis.

1. Introduction

The success of extended X-ray-absorption fine-structure (EXAFS) spectroscopy as a quantitative structural analysis technique is due, in part, to the availability of high-quality, user-focused software for calculation of theoretical photoelectron scattering functions for use in the analysis of the EXAFS data. One of the leading packages in this area is *FEFF* (Rehr & Albers, 2000). The success of *FEFF*'s EXAFS theory is due, in part, to the clear separation of the theory calculation from its use in data analysis software. *FEFF* is run as an independent process with output written to disk. This disk output is then used as fitting standards for the EXAFS analysis. This separation of theory from analysis has resulted in a thriving ecosystem of data analysis packages using *FEFF*, with over a dozen distinct efforts in circulation at this time (Vaarkamp *et al.*, 1994; George & Pickering, 1995; Kuzmin, 1995; Sanchez Del Rio *et al.*, 1995; Klementev, 2000; Curis, 2001; Newville, 2001; Ravel & Newville, 2005; Taguchi *et al.*, 2005; Webb, 2005; Krayzman *et al.*, 2009; Michalowicz *et al.*, 2009; Meneghini *et al.*, 2012; Stötzel *et al.*, 2012; Newville, 2013; Delgado, 2014; Figueroa & Prestipino, 2016).

Recently, *The Feff Project* (FEFF, 2017) made a version of *FEFF8* (Ankudinov *et al.*, 1998) available under a license that permits modification and redistribution of the code. This version is called *FEFF8L*, where the *L* indicates that this 'light' version of *FEFF* is intended specifically for EXAFS applications. It offers all of *FEFF*'s functionality related to EXAFS, but without tools for calculating X-ray-absorption near-edge structure (XANES) or other spectroscopies. This EXAFS-oriented code base is now freely available from an on-line source code repository (FEFF8L, 2017) and is freely redistributable.



© 2017 International Union of Crystallography

The goal of the *FEFF8L* project is to reconfigure this EXAFS-oriented code base to make integration of its theory into analysis software easier and more expressive. The current model – system calls to the *FEFF* executable which produces ASCII output files that must then be parsed by the analysis software – is cumbersome and complicates the development of new strategies of data analysis and visualization (Ravel, 2013, 2015).

To review how *FEFF* has been used for EXAFS data analysis, Fig. 1 shows a schematic of the flow of data through *FEFF*.

(1) The Cartesian coordinates of a cluster of atoms and the parameters of the calculation are provided as the input data. Traditionally, *FEFF* obtains these data by parsing a specially structured ASCII input file.

(2) The positions and species of the atoms in the cluster are used to create a muffin-tin potential surface along with energy-dependent self-energies (Rehr & Albers, 2000) computed by one of several models. These potentials are used to compute the effective, complex, energy-dependent, photoelectron scattering functions for each kind of atom in the cluster.

(3) The positions and species in the cluster are used to enumerate an exhaustive list (Zabinsky *et al.*, 1995) of single- and multiple-scattering geometries.

(4) For each scattering geometry considered in the EXAFS analysis, a data table is generated using the photoelectron scattering functions.

This data table is the content of *FEFF*'s output ASCII files and is the input to each of the EXAFS analysis programs, serving as the theoretical fitting standard.

Step 2, the calculation of the muffin-tin potentials, is the main focus of this manuscript. The calculation begins by placing neutral, spherical atoms computed with a relativistic formalism (Desclaux, 1973) at the Cartesian coordinates indicated in *FEFF*'s input data. In general, these neutral

atoms overlap. The atoms are then reduced in size (Norman, 1976) such that the smaller spheres just touch (or possibly slightly overlap) the boundaries of the Wigner–Seitz cells (Wigner & Seitz, 1933) formed from the input coordinates. This results in a muffin-tin of charged, spherical atoms with the remaining electron density distributed uniformly throughout the volume. From these atoms, the scattering potentials are computed, which are then used to compute the complex scattering function needed for evaluation of the EXAFS equation in the fit to the EXAFS data.

It is not obvious that this model of the potential surface is adequate for EXAFS analysis. Chemical bonding implies a transfer of charge between atoms and an asymmetric distribution of charge on atoms and throughout the volume, features that neutral spherical muffin tins lack. Nonetheless, this formalism has been highly successful for EXAFS analysis.

Such a simple approach to the potential surface proved inadequate for computation of the X-ray absorption near-edge structure (XANES) and other related spectroscopies, however. In *FEFF8L*, self-consistent potentials were introduced and used in all later versions. The self-consistency loop begins by computing potentials as described above. From those potentials, the electronic densities of state (DOS) are computed for all the atoms in the cluster. These DOS are integrated in energy until all the charge contained within the spheres is accounted for. This resets the Fermi energy and redistributes charge among the atoms. The potentials are calculated anew, the DOS are computed from the new potentials, and the integration is again performed. This loop continues to self-consistency, *i.e.* until the Fermi energy and the charge distribution stop changing. While this self-consistency loop is essential for the XANES calculation (Ankudinov *et al.*, 1998), its impact on EXAFS analysis has never been considered quantitatively.

2. Open source theoretical EXAFS standards

The work-flow described above, along with its associated input and output of data, is handled by *FEFF* in a way that made good sense in the mid-1990s, when *FEFF6* was developed, when computer speed and memory were limited, and when compilers were less sophisticated. Twenty years later, this work-flow merits reconsideration. The code contributed by *The Feff Project* is written¹ in Fortran, in a style that is mostly Fortran77 with some features of Fortran90.

The *FEFF8L* project has accomplished several goals for this code base:

(1) Combine the parts of *FEFF* concerned with the calculation of the potentials (*POT*, in the nomenclature of the *FEFF* code base) and the photoelectron scattering factors (*XSPH*) into a single stand-alone library which generates the potentials data used to compute the contributions to the EXAFS of individual scattering paths.

¹ Any mention of software programming languages is for information only; it does not imply recommendation or endorsement by the National Institute of Standards and Technology.

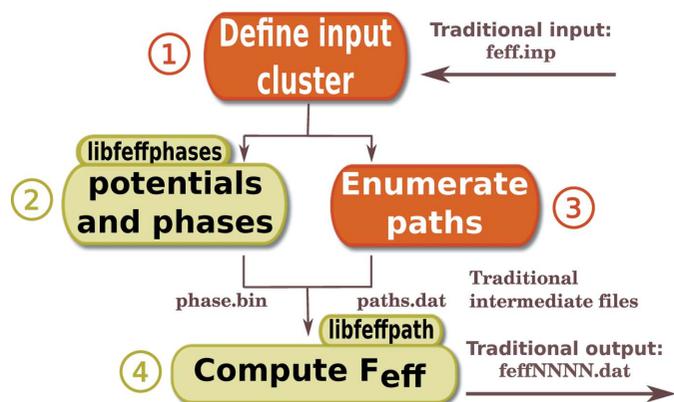


Figure 1
FEFF flow chart. The traditional flow of data between the parts of *FEFF* is by files read from and written to disk. The libraries described in §2 perform chores 2 and 4. With the exception of the file containing the phase information, the entire traditional input/output structure of *FEFF* is discarded by *FEFF8L*. Chores 1 and 3 are not the topic of this manuscript.

(2) Combine the parts of *FEFF* associated with computing the contribution to the EXAFS from a specific scattering path (*GENFMT* and *FF2X*) into a single stand-alone library which returns a data table containing the contributions to the EXAFS of the individual scattering path.

(3) For each of these stand-alone libraries, provide an entry point in the form of Fortran code which can be compiled against a program written in Fortran. For each of these Fortran entry points, provide a wrapper library written in C which provides a data structure and a clear application program interface for compilation against programs written in C, C++ or other compiled languages. For each of these C wrappers, provide interfaces for dynamic languages, such as Python or Perl.

(4) Verify that all code compiles with few or no warnings and triggers no warnings when used with a memory error detector. Where possible, replace the use of common blocks with long argument lists, making it easier to develop C wrappers for the components of *FEFF*.

(5) Provide a robust unit testing system to validate changes made to the code by comparing *FEFF*'s output against calculations made using the code as originally contributed.

The library from item (1) uses information about the positions and types of the atoms in the cluster along with several parameters of the calculation. It is used to write a single file containing a description of the atomic potentials that are later used to compute the contributions to the EXAFS signal from scattering paths. This Fortran library is wrapped by a C library called `libfeffphases`, which offers an application program interface and a data structure that encapsulates *FEFF*'s calculation of potentials and scattering functions. Bindings to the C library for dynamic languages are readily implemented. Thus, computation of photoelectron scattering factors can be tightly integrated into any application written in almost any programming language.

Similarly, the library from item (2) is wrapped by a C library called `libfeffpath` and also has bindings for dynamic languages. Using the file with information about the photoelectron scattering factors written by `libfeffphases` and a description of a scattering geometry, this library generates the data table containing the theoretical fitting standard for that scattering geometry without the need for a system call or for writing output files to disk.

An example of using `libfeffpath` in a C program to compute the contribution from a scattering path is shown in Fig. 2. The header file defining the *FEFF* path data structure is included at line 1. At lines 4 through 6, memory is allocated for the data structure. At line 7, the file containing the data for copper metal is specified. This is the output file from `libfeffphases`. The path calculated in this example is a triangular double scattering path that includes an atom in the second coordination shell and an atom in the first. The degeneracy of the path is set at line 10. At lines 11 and 12, the positions of the scattering atoms and their unique potential indices are specified. The `add_scatterer` method also keeps track of how many legs are in the path, storing that and other geometric information in the data structure. At line 13, the

```

1 #include "feffpath.h"
2 int main() {
3     long ret;
4     FEFFPATH *path;
5     path = malloc(sizeof(FEFFPATH));
6     ret = create_path(path);
7     strcpy(path->phpad, "phase.pad");
8
9     /* fourth path (triangle) in copper */
10    path->degen = 48.0;
11    ret = add_scatterer(path, 0, 0, 3.61, 1);
12    ret = add_scatterer(path, 1.805, 0, 1.805, 1);
13    ret = make_path(path);
14
15    /* examine error codes (ret), then */
16    /* do something with this path */
17
18    cleanup(path);
19    return 0;
20 }
```

Figure 2

A tiny C program demonstrating the use of the `libfeffpath` library to compute the contribution from a double scattering path in copper metal.

contribution from the path is computed and stored in the data structure and is available for use.

Wrappers to the C libraries for languages like Python and Perl handle chores like memory management and string management more elegantly than C. Code equivalent to Fig. 2 in a high-level language is typically shorter and simpler to program correctly.

This simple example indicates how easily computation of the theoretical standards can be integrated into an EXAFS analysis program. A program must, in some way, keep track of the scattering geometries represented in the fitting model. This might be done using *FEFF*'s pathfinder or it could use positions determined from molecular dynamics, reverse Monte Carlo or any other kind of numeric procedure. Once armed with the coordinates of the scattering atoms in a path, the contribution from that path can be generated as needed.

3. Testing self-consistency on a curated set of standards

The unit testing framework, item (5) above, was implemented using *Larch*, an open-source toolkit for processing and analyzing scientific data (Newville, 2013), and the *nose* (Pellerin, 2017) unit testing framework to validate work performed on the code by verifying that the output from the newly modified code is unchanged compared with the output of *FEFF8L* in its original state. Many tests are made, including direct comparisons of *FEFF*'s output and examination of the results of fits made to real EXAFS data. By running this test suite frequently during development, changes to the code are made safely.

This testing framework was extended to investigate the effect of changes in the theoretical model on the results of EXAFS fitting. For a curated set of measured data, EXAFS analysis was performed using *FEFF6* (Zabinsky *et al.*, 1995), *FEFF8L* without the use of self-consistent potentials (SCP) and *FEFF8L* with SCP calculated with increasing SCP radii. The SCP radius indicates the number of atoms in the cluster

to include in the SCP calculation. In these tests, an increasing sequence of radii are used to verify that the effect of self-consistency on the EXAFS is properly converged. The smallest radius just enclosed the nearest-neighbor distance, while the largest enclosed four or more coordination shells. In each case, *FEFF*'s default model for the energy-dependent Hedin–Lundqvist (Zabinsky *et al.*, 1995) self-energy is used.

The set of standards includes: copper foil, NiO, FeS₂ (pyrite), UO₂ (uraninite), BaZrO₃, C₁₀H₁₅Br (bromoadamantane) and uranyl hydrate in solution. Copper, NiO (Ravel, 2015) and FeS₂ are examples of well ordered, dense, crystalline solids for which SCP and charge transfer may not play a big role. UO₂ (uraninite) (O'Loughlin *et al.*, 2003) is a well ordered, *f*-electron system. BaZrO₃ was suggested (Haskell *et al.*, 1995) as a test of whether shortcomings in *FEFF*'s potential model could be accommodated by multiple energy shift parameters in the fitting model. Bromoadamantane is a molecule and its EXAFS data include a measurable contribution from hydrogen scatterers. Finally, uranyl hydrate (Kelly *et al.*, 2002) is a molecule in solution, an *f*-electron system, and a structure with two extremely short (~1.79 Å), covalent, oxygenyl ligands. In this manuscript, only the results for NiO and UO₂ will be presented in detail. Full fitting results for all seven standards and all theory conditions are presented in the supporting information.

In each case, the measured absorption cross sections, $\mu(E)$, were processed using the *ATHENA* program (Ravel & Newville, 2005). In each case, the edge energy was set during background removal to be the inflection point of the main rising edge of the XANES data. While this may not be the most appropriate choice of edge energy (E_0) in each case, it was important for this exercise that the data be processed in a manner that was consistent and easily reproducible.

As with any EXAFS analysis, we start with the EXAFS equation for the contribution from a single scattering path as a function of photoelectron wavenumber k ,

$$\chi_{\Gamma}(k) = \frac{NS_0^2 F_{\Gamma}(k) \exp(-2k^2\sigma_{\Gamma}^2) \exp[-2R_{\Gamma}/\lambda(k)]}{2kR_{\Gamma}} \times \sin[2kR_{\Gamma} + \delta_{\Gamma}(k)]. \quad (1)$$

Here, Γ is an index counting over all scattering paths considered in the fit. $F_{\Gamma}(k)$ and $\delta_{\Gamma}(k)$ are the amplitude and phase of the photoelectron scattering function and are computed by *FEFF*, as is $\lambda(k)$, the photoelectron mean free path. The remaining terms, *i.e.* N , the path degeneracy; S_0^2 , the passive electron reduction factor (Li *et al.*, 1995); R , the scattering path length; and σ^2 , the mean square variation in scattering path length, are all parameterized using the variables of the fit. Additionally, a parameter adjusting the edge energy value, ΔE_0 , is used to modify the zero of photoelectron wavenumber k when fitting theory to data. Finally, all paths used in the analysis are summed,

$$\chi_{\text{theory}}(k) = \sum_{\Gamma} \chi_{\Gamma}(k), \quad (2)$$

and $\chi_{\text{theory}}(k)$ is compared with the measured $\chi(k)$.

This set of standards is chosen because each is of a well known structure. For each, a successful fitting model can be made using the known structure as prior knowledge of the fit. Without this level of prior knowledge, it would be difficult to distinguish systematic uncertainty in the fit due to an inadequate model of the local structure from systematic uncertainty due to shortcomings of the underlying theoretical model. Thus, this set of well characterized standards allows specific evaluation of the effect on the fits of changes to the theoretical model.

3.1. NiO

The EXAFS data shown in Fig. 3 are for the nickel oxide standard shown by Ravel (2015). A simple fitting model accounts for the structure of cubic NiO. The model includes single and multiple scattering to the distance of the oxygen scatterers in the fourth coordination shell. A volumetric lattice expansion coefficient was used to vary the nominal lattice constant $a = 4.1684$ Å (Cairns & Ott, 1933) and to model all the path length parameters with a single variable. An S_0^2

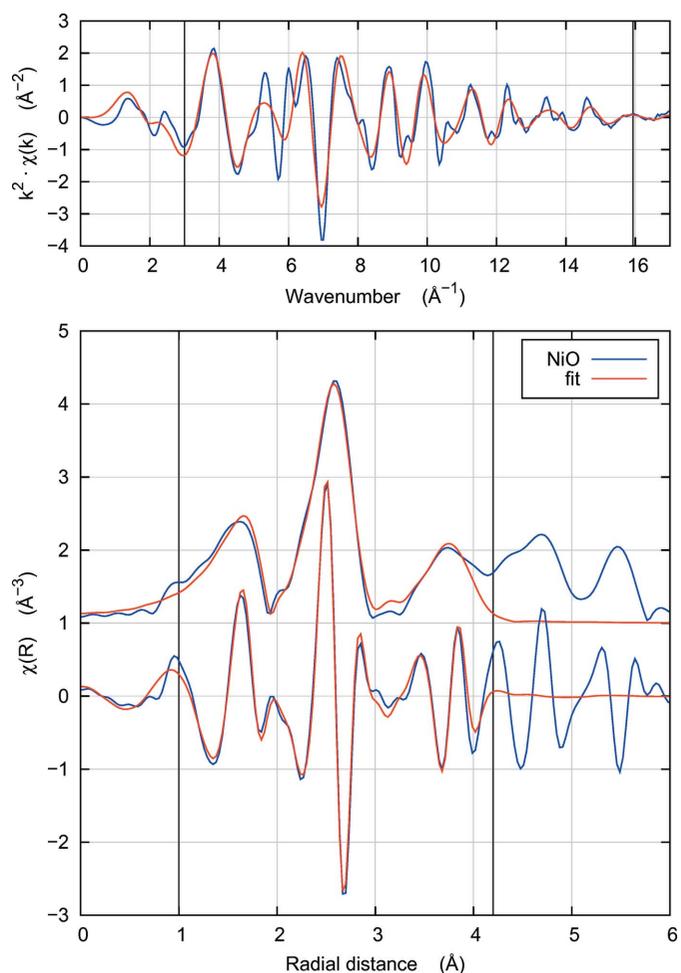


Figure 3 Fit to NiO using *FEFF8L* and a radial cluster size of 4.7 Å (column 7 in Table 1) to compute self-consistent potentials. The vertical lines denote the ranges of the Fourier transform in k and of the fit in R . The structural model and fitting parameters explained by Ravel (2015) were used. This figure is representative of all the fits reported in Table 1, the plots of which are nearly indistinguishable.

Table 1

Fit results for NiO using the structural model described by Ravel (2015) and theory calculated using *FEFF6* and *FEFF8L* with and without SCP, which were computed with differently sized clusters (Ankudinov *et al.*, 1998) for the full multiple scattering matrix.

1σ uncertainties on the trailing digits are computed from the diagonal elements of the covariance matrix of the fit and are given in parentheses. χ_v^2 is the fitting metric reduced by the unused information content of the data. \mathcal{R} -factor represents a percentage misfit.

	<i>FEFF6</i>	<i>FEFF8L</i> , no SCP	SCP, $R = 2.5 \text{ \AA}$	SCP, $R = 3.7 \text{ \AA}$	SCP, $R = 4.7 \text{ \AA}$
S_0^2	0.71 (5)	0.68 (5)	0.71 (4)	0.71 (4)	0.71 (4)
a	(\AA) 4.173 (8)	4.173 (8)	4.168 (8)	4.164 (8)	4.164 (8)
$\sigma^2(\text{O})$	(\AA^2) 0.0044 (12)	0.0047 (13)	0.0047 (12)	0.0046 (12)	0.0047 (12)
$\sigma^2(\text{Ni})$	(\AA^2) 0.0055 (6)	0.0053 (6)	0.0055 (6)	0.0056 (6)	0.0055 (6)
E_0	(eV) -1.2 (5)	2.5 (6)	-7.3 (5)	-7.9 (5)	-7.3 (5)
χ_v^2	1347	1467	1379	1324	1378
\mathcal{R} -factor	0.022	0.023	0.022	0.021	0.022

parameter was used as an overall amplitude and an E_0 parameter was used to align the energy grid of the data with the theory. Finally, a σ^2 parameter was used for each of the four single-scattering paths corresponding to the four coordination shells. σ^2 for the multiple-scattering paths were parameterized in terms of the single-scattering σ^2 parameters in the manner explained by Hudson *et al.* (1996).

That fitting model was replicated and the fit was repeated using the sequence of theoretical calculations explained above. One example of a fit is shown in Fig. 3 and the results of the full sequence of fits are given in Table 1.

The quality of the fit to the NiO data as measured by the standard statistical parameters, reduced χ^2 and R -factor, shows scant variation across the sequence of theoretical models. Similarly, the values of the fitting parameters, other than E_0 , are unchanged outside of their error bars.

E_0 was the only fitting parameter affected by the details of the theory. The self-consistency of the potentials and the associated charge transfer move the *calculated* energy threshold. Since E_0 is best interpreted as a parameter which aligns the energy grid of the theory with the energy grid of the data, the addition of self-consistency is reasonably expected to change the fitted value of E_0 , although we find the E_0 value to have no correlation with SCP radius. In the case of NiO, SCP resulted in a fitted E_0 shift farther from 0, although this result should not be over-interpreted. This is not a statement about where in the data the threshold energy is found – that is certainly near the inflection point, as discussed above – rather it is an indication of the difficulty of the theoretical task of computing the threshold position. The fitted value for E_0 corrects that shortcoming in the theory.

A few more words regarding the interpretation of E_0 are merited in the context of this exercise. As a numerical fitting parameter, E_0 is always highly correlated with the various ΔR parameters because both predominantly affect the phase terms in the EXAFS equation. In all the examples discussed here and in the supporting information, the data range is long which helps to distinguish the effects of E_0 and ΔR in the fit. In cases where the data range is limited or structural disorder is high, distinguishing the phase parameters is much more challenging. The variability in fitted E_0 among the different

theoretical models does not mean that the value of the fitted E_0 is unimportant. Like all fitting values, the value E_0 must be defensible in any fit.

3.2. UO_2

The EXAFS data on uraninite (UO_2) shown in Fig. 4 are the data presented by O'Loughlin *et al.* (2003). This is an interesting test case for the theoretical model as it involves covalent bonding of oxygen to uranium, an f -electron metal.

Following the lead of O'Loughlin *et al.* (2003), *FEFF* was run using the structure of crystalline UO_2 (Wyckoff,

1964). Single-scattering paths from the first three coordination shells and a variety of significant multiple-scattering paths are included in the fitting model. The parameterization of the EXAFS equation given in Table 1 of O'Loughlin *et al.* (2003) is used here.

An S_0^2 parameter was used as an overall amplitude and an E_0 parameter was used to align the energy grid of the data with the theory. R and σ^2 parameters are floated for each single-scattering path. The nominal R values for the first three shells in the fluorite structure of UO_2 (Wyckoff, 1964) are 2.368 \AA , 3.867 \AA and 4.534 \AA . R values for the multiple-scattering paths are related to the single-scattering paths by geometry while σ^2 parameters are related as explained by Hudson *et al.* (1996).

As with the fit sequence to NiO, there is little variation in statistical parameters across this sequence of fits. Values for R and σ^2 do not change outside their error bars. There is a difference in the best-fit value of N_{U} , the coordination number of the second-shell uranium scatterers, between *FEFF6* and *FEFF8L*. This is nominally 12 but, as explained by O'Loughlin *et al.* (2003), is allowed to float to model vacancy. As is common for the determination of a coordination number in EXAFS analysis, the uncertainty in N_{U} is quite large, larger than the variation between the fitting models, and correlated with both S_0^2 and σ_{U}^2 . While this change in N_{U} between *FEFF6* and *FEFF8L* might change the interpretation of these results, the difference is, in fact, well within the level of uncertainty.

Again, the fitted value of E_0 changes with SCP. With SCP, E_0 becomes closer to 0, as might be anticipated by this improvement to the theory. As shown in the supporting information, this same behavior was true for uranyl hydrate, but was not seen for any other material tested. Again, there was no dependence of E_0 on SCP radius.

4. Conclusion

In this set of standards (see the supporting information for complete details) there is little dependence of the statistical parameters on the theoretical model, sometimes χ_v^2 and \mathcal{R} -

Table 2

Fit results for uraninite (UO₂) following the structural model of O’Loughlin *et al.* (2003) and using theory calculated with *FEFF6* and *FEFF8L* with and without SCP, which were computed with differently sized clusters (Ankudinov *et al.*, 1998) for the full multiple scattering matrix.

1σ uncertainties on the trailing digits are computed from the diagonal elements of the covariance matrix of the fit and are given in parentheses. χ_r² is the fitting metric reduced by the unused information content of the data. R-factor represents a percentage misfit.

	<i>FEFF6</i>	<i>FEFF8L</i> , no SCP	SCP, R = 3 Å	SCP, R = 5 Å	SCP, R = 6 Å
S ₀ ²	0.87 (11)	0.84 (11)	0.84 (10)	0.84 (10)	0.84 (10)
N _U ²	11.4 (4.8)	9.3 (4.2)	9.2 (3.8)	9.2 (3.7)	9.3 (3.7)
R(O ₁) (Å)	2.34 (1)	2.34 (2)	2.33 (1)	2.33 (1)	2.33 (1)
R(U ₂) (Å)	3.86 (2)	3.86 (3)	3.86 (3)	3.86 (2)	3.86 (2)
R(O ₃) (Å)	4.48 (4)	4.51 (6)	4.52 (5)	4.52 (5)	4.52 (5)
σ ² (O ₁) (Å ²)	0.0094 (21)	0.0087 (22)	0.0089 (21)	0.0089 (21)	0.0089 (21)
σ ² (U ₂) (Å ²)	0.0049 (25)	0.0038 (27)	0.0039 (25)	0.0039 (25)	0.0039 (25)
σ ² (O ₃) (Å ²)	0.0106 (44)	0.0106 (62)	0.0098 (51)	0.0097 (51)	0.0097 (51)
E ₀ (eV)	4.9 (1.4)	8.2 (1.5)	1.7 (1.3)	1.7 (1.3)	1.7 (1.3)
χ _r ²	166	188	170	169	169
R-factor	0.016	0.018	0.016	0.016	0.016

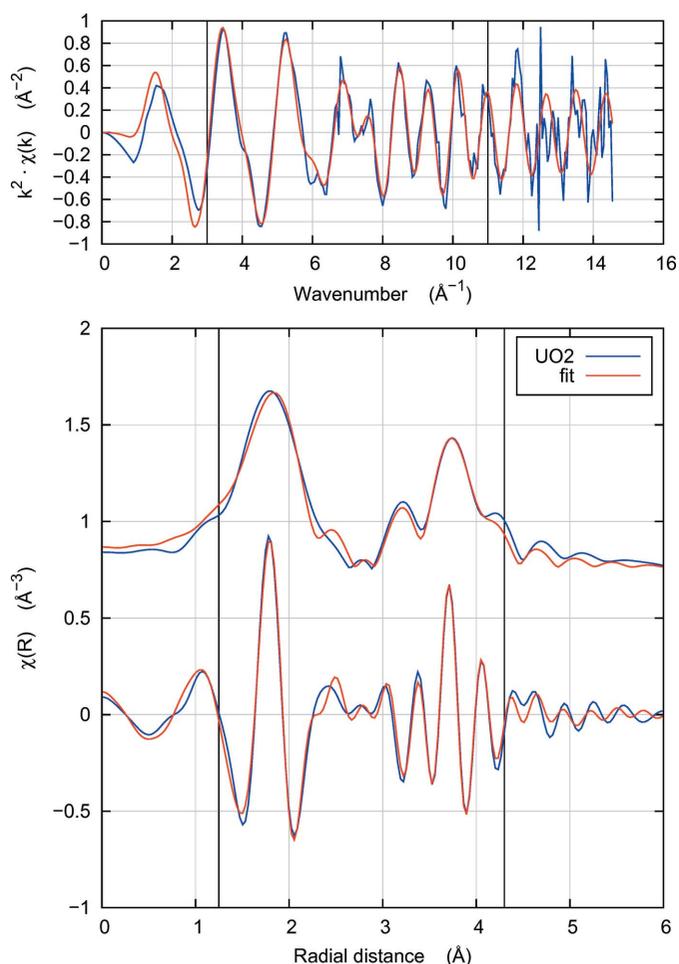


Figure 4
Fit to UO₂ using *FEFF8L*, a radial cluster size of 6 Å (column 7, Table 2) to compute self-consistent potentials. The vertical lines denote the ranges of the Fourier transform in *k* and of the fit in *R*. The structural model and fitting parameters follow the example of O’Loughlin *et al.* (2003). This figure is representative of all the fits reported in Table 2, the plots of which are nearly indistinguishable.

factor are smaller with *FEFF8L*, sometimes with *FEFF6*. Similarly, there is little variation in the fitted values of variable parameters with and without SCP. The sole exception is with the E₀ parameter. Because this parameter is used to align the energy grids of the data and theory, it is sensitive to the position of the threshold energy in the calculation. This naturally changes during the self-consistency loop. There is very little further variation in E₀ as the SCP radius is increased beyond its smallest value. This is a useful result. Because the SCP calculation is time-consuming for a large radius, this result offers confidence that a small radius is adequate for EXAFS analysis. Only in the case of the

uranium compounds did *FEFF8L* result in an E₀ value closer to 0. Whether this is a meaningful result for *f*-electron systems is a topic for further study.

This work on nickel oxide, uraninite and the other standards shown in the supporting information demonstrates that the use of self-consistent potentials has scant impact on EXAFS analysis. This can be understood in general terms by considering the kinetic energy of the photoelectron. Near the edge, the photoelectron has very little kinetic energy (*i.e.* the difference between the energy of the incident photon and the binding energy of the deep core electron). The propagation of a photoelectron with low kinetic energy depends strongly upon the details of the potential surface. It is for this reason that self-consistency is so important (Ankudinov *et al.*, 1998) for a XANES calculation. In the EXAFS region, the photoelectron has much more kinetic energy and so is relatively unaffected by the details of the potential surface. This is the reason that the muffin-tin potential has proven so successful in EXAFS analysis as an approximation for the actual potential surface. The change to the calculated potential surface introduced by SCP is small compared with the photoelectron kinetic energy in the EXAFS regions, thus has little impact on the analysis of the EXAFS.

FEFF6 has long been used (Ravel & Newville, 2005) for fitting standards in EXAFS analysis. This is reasonable practice. Transitioning EXAFS analysis software to use *FEFF8L* is attractive in order to leverage the application program interface developments described in §2. When using SCP for EXAFS analysis, this work demonstrates that a small cluster radius, enough to include the first coordination shell, is sufficient.

Note that the *FEFF8L* effort does not include a tool for organizing the input data specifying the configuration of the input cluster, although *libfeffphases* provides a data structure for this information. How the cluster configuration is specified is a chore best left to the specific analysis application. A conventional *FEFF* input file could be parsed or the

configuration could be specified in some other manner defined by the software or the application.

The effect of other aspects of the theory on the EXAFS analysis could be subjected to this sort of rigorous testing. Some examples might include the exchange potential model (Rehr & Albers, 2000), the self-energy model (Kas *et al.*, 2007) or certain parameters of the calculation such as the grid of the radial integration or the wavenumber grid of the output data table.

Acknowledgements

Use of the National Synchrotron Light Source, Brookhaven National Laboratory, was supported by the US Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-98CH10886. We thank Dr Alessandra Leri, Marymount Manhattan College, for providing the bromoadamantane sample. We thank two anonymous reviewers at the 16th International Conference on X-ray Absorption Fine Structure for their helpful comments on an early draft of this manuscript.

Funding information

JJR and JJK were supported by DOE BES Grant DE-FG02-97ER45623.

References

- Ankudinov, A. L., Ravel, B., Rehr, J. J. & Conradson, S. D. (1998). *Phys. Rev. B*, **58**, 7565–7576.
- Cairns, R. W. & Ott, E. (1933). *J. Am. Chem. Soc.* **55**, 527–533.
- Curis, E. (2001). PhD thesis, Université Paris-Sud, France.
- Delgado, M. (2014). *EXAFSArchitect*, <http://sourceforge.net/projects/exafsarchitect/>. Accessed 3 April 2017.
- Del Rio, M. S., Chaboy, J. & Djerbi, R. (1995). *Comput. Chem.* **19**, 59–61.
- Desclaux, J. P. (1973). *At. Data Nucl. Data Tables*, **12**, 311–406.
- FEFF (2017). *The FEFF Project*, <http://feffproject.org/>. Accessed 3 April 2017.
- FEFF8L (2017). *FEFF8L: Open Source theoretical EXAFS calculations*, <https://github.com/xraypy/feff85exafs/>. Accessed 3 April 2017.
- Figuroa, S. J. A. & Prestipino, C. (2016). *J. Phys. Conf. Ser.* **712**, 012012.
- George, G. & Pickering, I. (1995). *EXAFSPAK: a suite of computer programs for analysis of X-ray absorption spectra*, <http://www-ssrl.slac.stanford.edu/exafspak.html>. Accessed 3 April 2017.
- Haskel, D., Ravel, B., Newville, M. & Stern, E. (1995). *Physica B*, **208–209**, 151–153.
- Hudson, E. A., Allen, P. G., Terminello, L. J., Denecke, M. A. & Reich, T. (1996). *Phys. Rev. B*, **54**, 156–165.
- Kas, J. J., Sorini, A. P., Prange, M. P., Cambell, L. W., Soininen, J. A. & Rehr, J. J. (2007). *Phys. Rev. B*, **76**, 195116.
- Kelly, S., Kemner, K., Fein, J., Fowle, D., Boyanov, M., Bunker, B. & Yee, N. (2002). *Geochim. Cosmochim. Acta*, **66**, 3855–3871.
- Klementev, K. (2000). *Nucl. Instrum. Methods Phys. Res. A*, **448**, 299–301.
- Krayzman, V., Levin, I., Woicik, J. C., Proffen, Th., Vanderah, T. A. & Tucker, M. G. (2009). *J. Appl. Cryst.* **42**, 867–877.
- Kuzmin, A. (1995). *Physica B*, **208–209**, 175–176.
- Li, G. G., Bridges, F. & Booth, C. H. (1995). *Phys. Rev. B*, **52**, 6332–6348.
- Meneghini, C., Bardelli, F. & Mobilio, S. (2012). *Nucl. Instrum. Methods Phys. Res. B*, **285**, 153–157.
- Michalowicz, A., Moscovici, J., Muller-Bouvet, D. & Provost, K. (2009). *J. Phys. Conf. Ser.* **190**, 012034.
- Newville, M. (2001). *J. Synchrotron Rad.* **8**, 322–324.
- Newville, M. (2013). *J. Phys. Conf. Ser.* **430**, 012007.
- Norman, J. G. (1976). *Mol. Phys.* **31**, 1191–1198.
- O’Loughlin, E. J., Kelly, S. D., Cook, R. E., Csencsits, R. & Kemner, K. M. (2003). *Environ. Sci. Technol.* **37**, 721–727.
- Pellerin, J. (2017). *nose: nicer testing for python*, <http://nose.readthedocs.io/>. Accessed 3 April 2017.
- Ravel, B. (2013). *J. Phys. Conf. Ser.* **430**, 012006.
- Ravel, B. (2015). *J. Synchrotron Rad.* **22**, 1258–1262.
- Ravel, B. & Newville, M. (2005). *J. Synchrotron Rad.* **12**, 537–541.
- Rehr, J. J. & Albers, R. C. (2000). *Rev. Mod. Phys.* **72**, 621–654.
- Stötzel, J., Lützenkirchen-Hecht, D., Grunwaldt, J.-D. & Frahm, R. (2012). *J. Synchrotron Rad.* **19**, 920–929.
- Taguchi, T., Ozawa, T. & Yashiro, H. (2005). *Phys. Scr.* **T115**, 205.
- Vaarkamp, M., Dring, I., Oldman, R. J., Stern, E. A. & Koningsberger, D. C. (1994). *Phys. Rev. B*, **50**, 7872–7883.
- Webb, S. M. (2005). *Phys. Scr.* **T115**, 1011.
- Wigner, E. & Seitz, F. (1933). *Phys. Rev.* **43**, 804–810.
- Wyckoff, R. G. W. (1964). *Crystal Structures*, Vol. 1–6. New York: Interscience.
- Zabinsky, S. I., Rehr, J. J., Ankudinov, A., Albers, R. C. & Eller, M. J. (1995). *Phys. Rev. B*, **52**, 2995–3009.