SIXPACK: a graphical user interface for XAS analysis

Samuel M. Webb*

Stanford Synchrotron Light Source, SLAC National Accelerator Laboratory, 2575 Sand Hill Road, Menlo Park, CA 94025, USA. *Correspondence e-mail: samwebb@slac.stanford.edu

SIXPACK, a graphical user interface that allows simple manipulation and analysis of X-ray absorption spectroscopy (XAS) data, is presented. The modules of SIXPACK allow users to (i) load, calibrate and average raw data files, (ii) perform background subtractions, (iii) perform principal component analysis and target transforms, (iv) perform least-squares fitting of data to standards and functions, (v) perform EXAFS fitting to FEFF phase and amplitude files and (vi) create single-scattering FEFF phase and amplitude files using a periodic table interface. The core of the XAS analysis routine uses IFEFFIT. Novel features of the program allow for the fitted correction of XANES spectra due to self-absorption effects in unknown matrices, which is particularly useful for analysis of geochemical and environmental systems. Data formats from many synchrotrons and the XAS Data Interchange (XDI) format are directly supported, and a generic file importer can be used for unknown data formats. SIXPACK is developed in Python, is installable across many operating systems and platforms, and is freely available with an open-source licence.

1. Introduction

As synchrotron facilities are inspiring new applications for XAS experiments, they are also reaching broader user bases than in the past. Many of these users are new to the methodology and physics of XAS analyses and bring highly complex materials requiring sophisticated data-analysis techniques. This has led to the need for simple data-analysis packages that are easy to use for beginners, yet powerful enough to tackle complex problems for advanced users.

IFEFFIT is a well tested, reliable, interactive scriptable library of EXAFS algorithms (Newville, 2001; Newville & Ravel, 2020). An additional attractive quality of IFEFFIT is that it can be used easily from high-level scripting languages, such as Python. In this fashion, easy-to-use graphical user interfaces (GUIs) can be built to aid in the analysis of XAS data. SIXPACK (Webb, 2005) is a GUI that is designed with this in mind, utilizing IFEFFIT as the analysis ‘kernel’ and supplementing it with many other routines for data input and other analysis.

While SIXPACK is used for visualizing data during the data-analysis process, it is not intended for making publication-quality graphs. Every graphical display in the program, however, has an ‘Export Plot Data to Clipboard’ function, which allows easy export of the data to the graphing program of the user’s choice. Data can also be saved at each step along the process into ASCII data files.

SIXPACK has been developed for use initially with the environmental and geochemical fields in mind, but has been shown to be useful in a wide range of XAS applications. SIXPACK is distributed as compiled code for Windows and Mac OSX platforms. Written in Python (version 2.7; http://
www.python.org), the code can also be used on Linux systems from the source code, as long as all dependencies are installed. Source code, compiled versions and documentation can be found at https://www.sams-xrays.com/ and are freely available with an open-source licence.

2. Program description

**SIXPACK** is divided into six basic modules, with each section of the GUI performing a specific task in the data-reduction and analysis process, as described below.

2.1. SamView

The SamView module is used to import raw data into the program and perform basic initial reduction tasks, such as averaging, energy calibration and deadtime corrections. The module attempts to recognize the headers of many different data formats, including SSRL ASCII and binaries, NSLS-I text formats and BSIF formats, ALS 10.3.2, PNC-CAT, GSECARS, DND-CAT, SRC and the XAS Data Interchange (XDI) format (Ravel et al., 2012), among others. If the data format is not recognized, a generic ASCII reader can be used, allowing the user to specify the identity of the data columns. Future expansion to other formats is welcomed, and user requests can be made via https://www.sams-xrays.com/. The module GUI provides an easy way to screen data files, eliminate problem channels from multi-element detector systems and perform most pre-processing procedures. Batch processing of data files is also supported, following the SSRL naming convention, where filenames matching the same XXX pattern in filenames such as `foo_XXX_YYY.dat` are averaged together automatically. Averaged files are saved in a two-column ASCII format, as well as an XDI format.

2.2. Background-removal module

The background-removal module performs unit step edge normalization of XANES data and the extraction of EXAFS from averaged data. The EXAFS $\chi(k)$ is isolated using the **IFEFFIT** routines developed by Matt Newville. Data can be edited with a deglitching dialogue to remove spurious data points, and the energy scale can be adjusted if needed after the previous averaging steps. Self-absorption corrections can be applied at this point, as long as the sample composition and experimental geometries are well known. The correction uses algorithms derived from Booth & Bridges (2005). The algorithm is focused primarily on correcting the effects of concentrated samples on the EXAFS portion of the spectrum, in contrast to the empirical correction that can be applied during linear combination fitting (see Section 2.4).

Background-processed data can be saved as $\mu(E)$, derivative spectra, $\chi(k)$ and radial distribution functions. Data are saved in a generic two-column ASCII format, as well as the XDI format. For the XDI format, in addition to preserving the original data header containing the pertinent beamline and data-acquisition parameters, a record of the values used in the background removal is also added such that the history of the file processing is preserved in the file. Batch processing of a series of data files with the same or similar parameters is available.

2.3. Principal component analysis module

The principal component analysis (PCA) module performs statistical analyses on a set of spectra to determine the number of ‘significant’ or principal components required to describe the variation in the data. PCA can also be used to cluster a set of sample spectra into groups of similar spectra. When data files are loaded into the module, they are interpolated on a consistent energy grid, by default 0.1 eV for XANES data or 0.05 Å\(^{-1}\) for EXAFS. The interpolation grid can be set to a custom value or turned off completely if all data files are already on the same energy spacing. The **scikit-learn** toolkit (Pedregosa et al., 2011) is used for PCA calculations and allows a larger number of spectra to be analysed in a single calculation compared with a simple singular value decomposition algorithm. After the PCA is completed, the data can be rotated using a varimax rotation, which is a change of coordinates that maximizes the sum of the variances of the squared PCA loadings. Thus, all the coefficients (squared correlation with factors) will be either large or near-zero, with few intermediate values. The goal is to generally associate each variable to at most one factor, simplifying the interpretation of the results.

A number of criteria can be used in determining the number of principal components. The PCA module supports traditional tests, such as a skew test or a simple examination of how much of the cumulative sample variance is explained by a certain number of components. These two tests are often fairly subjective and require a fair amount of personal bias by the investigator. A fairly robust, yet empirical, statistical parameter developed by Malinowski, called the IND function, can also be used. This function reaches a minimum at the number of principal components (Malinowski, 1977).

The PCA results can also be used with target transformations. The target-transformation procedure uses the principal components defined by the sample set, and attempts to fit a reference compound with the set of the principal component vectors. This essentially projects the spectrum from the reference compound into the vector space that is defined by the spectra in the data set. If the target reference vector lies in this component space, then the reference compound is a good choice of a reference spectrum for further analysis, such as linear combination fitting. The quality of the target transform is given by three parameters: $\chi^2$, $R$ factor and the SPOIL value (Malinowski, 1978). The SPOIL value typically ranges from 0 to 1.5 for an excellent target, 1.5 to 3 for a good target, 3 to 4.5 for a fair target and 4.5 to 6 for a poor target; targets with values greater than 6 are unacceptable (Manceau et al., 2002).

To assist in the process of screening reference compounds, a library file of compounds can be created from which the module will perform a batch series of target transforms and report the quality of fit outcomes in a table.

The PCA module provides an interface to plot the individual sample loadings of two (or three) eigenvectors for the PCA data set. This allows the identification of important
sample set clusters and/or of extreme values, and thus sample spectra, that represent the closest end-members in the data set sample space.

2.4. Linear combination fitting

Fitting experimental EXAFS data to calculate bond distances and coordination numbers becomes increasingly difficult in systems that are comprised of more than a few components. In the natural environment, few elements will exist as pure phases owing to chemical and physical inhomogeneities in samples such as soils and sediments. Thus it becomes important to be able to examine the sample composition as a linear combination of a series of standard components, or references. This is accomplished through the linear least-squares fitting module of SIXPACK.

The fitting module of SIXPACK allows the user to fit data to any number of compounds and/or functions. Functions included at this time are linear and quadratic forms, exponential and logarithmic forms, arctangents and error functions, and Gaussian and Voigt peak shapes. Variables in the functional forms may be floated or fixed in the fitting process. Constraints on the proportions of reference compounds, such as non-negativity and sum-to-one constraints, may also be applied. Fitting can be performed on the data itself, or on the first, second or third derivatives of the data in cases where derivative fitting may lead to more sensitive species determination. The fitting routine is generic and not specific to X-ray absorption spectra, requiring only a two-column ASCII file as input. The fitting routines may be applied to other types of data sets as well, including X-ray diffraction and UV–visible spectroscopy. Data files are interpolated on the same energy grid on loading into the module: on a 0.1 eV grid for XANES and a 0.05 Å\(^{-1}\) grid for EXAFS. The default grid can be user-defined or the interpolation turned off if the energy grids of the sample and references are identical.

A typical problem when performing least-squares fitting is choosing an appropriate set of standard reference compounds, or basis set. PCA, as in Section 2.3, can be used to determine the number of possible components and, through target transform, which standards are appropriate. Often, the requirements for a well determined PCA cannot be met by a sample set, and other methods need to be applied outside of mere intuition. SIXPACK provides two alternative methods to assist in determining the appropriate basis set. First, the matrix-fit routine performs fits to the data set with the list of reference compounds in a full combinatorial sense, reporting the R-factor figure of merit for each fit. For \( n \) references, the matrix-fit routine computes \( 2^n - 1 \) fits, which for even small libraries can lead to long times for the full combinatorial

<table>
<thead>
<tr>
<th>Fit type</th>
<th>% as ( \delta )-MnO(_2)</th>
<th>% as MnCl(_2) (aq)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission</td>
<td>88</td>
<td>9</td>
</tr>
<tr>
<td>Fluorescence</td>
<td>77</td>
<td>18</td>
</tr>
<tr>
<td>SA-corrected fluorescence</td>
<td>89</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Fit type</th>
<th>% as ( \delta )-MnO(_2)</th>
<th>% as MnCl(_2) (aq)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission</td>
<td>88</td>
<td>9</td>
</tr>
<tr>
<td>Fluorescence</td>
<td>77</td>
<td>18</td>
</tr>
<tr>
<td>SA-corrected fluorescence</td>
<td>89</td>
<td>10</td>
</tr>
</tbody>
</table>

Figure 1

Fit qualities of biogenic manganese oxide to transmission reference compounds. (a) Fit of transmission sample data. (b) Fit of fluorescence sample data. (c) Fit of self-absorption-corrected fluorescence data.
matrix to be calculated. A second option, the cycle fit, is also available (Kim et al., 2013). Cycle fitting will initially fit the experimental spectrum with all \( n \) one-component fits for a set of \( n \) references, and report the associated \( R \) factor. The best-fit reference is then selected as the first component for the next cycle with the remaining \( n - 1 \) references. This cyclic process is repeated, adding the next \( y \)th reference with the best-fit \( R \) factor and then performing fits with the remaining \( n - y \) references, until adding another reference component no longer makes a significant improvement to the figure of merit for fitting. Significance could be considered a reference that must be >10% of the total fit, or one that must cause the \( R \)-factor figure of merit to decline by >10%. Generally, one will converge on a best fit much faster than a full combinatorial matrix fit, with similar best-fit results.

A novel feature of the least-squares fitting module is a parameterized correction for self-absorption (SA) or over-absorption effects in XANES spectra. This option applies an empirical correction factor derived from a simplified expression of the effect of SA in the sample on the observed data. Assuming an infinitely thick sample at 45° incidence and 45° detection (bisecting geometry), the XAS fluorescence measurement simplifies to

\[
I_f/I_0 = \frac{\mu_s(E)}{\mu_s(E) + \mu_m}.
\]

where \( \mu_s(E) \) is the rapidly changing absorption coefficient of the sample (i.e. the XANES or EXAFS) due to the element of interest, and \( \mu_m \) is the slowly varying absorption coefficient from other elements in the matrix. Since the desired quantity is the unimpacted absorption coefficient, \( \mu_s(E) \), we can solve for this term knowing the measured value of \( I_f/I_0 \) (represented by \( R \)), giving

\[
\mu_s(E) = \frac{f R \mu_m}{(1 - f R)}.
\]

where \( f \) is a scaling factor that accounts for normalization to a unit step height. In environmental or natural samples, the absorption coefficient of the matrix is largely unknown. Thus, in order to apply a correction, we can fit the data to an expression parameterized by two variables that characterize the net effect of the artefact: \( \mu_m \) and the scaling factor \( f \). These parameters modify the fluorescence data of the sample, where large values of either parameter imply no effect on the sample. An iterative process fits these SA variables to the former fit, followed by a new fit to the SA-corrected data. After several iterations, the two fits will converge if a correction factor is needed. An example of this procedure is given below.

2.4.1. Least-squares fitting with self-absorption correction. This example illustrates the self-absorption (SA) correction in SIXPACK. A sample of a homogeneous biogenic manganese oxide was collected in both fluorescence and transmission geometries on BL4-3 at SSRl using a harmonic rejection mirror set to a 9 keV cutoff energy. The transmission data can be fitted as a combination of manganese(II) (\( \text{MnCl}_2 \) in aqueous solution) and manganese(IV) (\( \delta\text{-MnO}_2 \)) reference compounds measured in transmission. These results are presented in Table 1 and Fig. 1(a). The same fit can be performed with the fluorescence data, resulting in a significantly different solution (Table 1 and Fig. 1b). The impact of SA in the fluorescence data is observed by a distortion of the pre-edge, white line and other features making the pre-edge transition appear larger and the white-line maximum smaller than observed in the transmission geometry. This artefact in the data, when fitted, results in the major component manganese(IV) being damped and appearing to be less
important to the overall fit. When the SA correction is applied, after ten iterations an improved fit is acquired from the self-absorption-impacted fluorescence data, with results that agree closely with the transmission fit (Table 1 and Fig. 1c).

2.5. **FEFF** single scattering interface module

*SIXPACK* contains a module to interactively create single-scattering **FEFF** phase and amplitude files. This provides a quick way to test scattering paths without building large complex **FEFF** models. The GUI consists of an easy-to-use clickable periodic table from which the scattering and absorbing elements are chosen. The user may control several of the standard **FEFF** parameters, and chooses a distance and geometry of the complex. The module is compatible with **FEFF** versions 6 through 8 (Ankudinov et al., 1998; Zabinsky et al., 1995; Kas et al., 2020).

![Figure 3](image1.png)

**Figure 3**
Screenshot of the variable-setup page of the EXAFS fitting module in **SIXPACK**, showing a model calculating coordination numbers of the two shells as a function of the percentage of the species present, and constraints on the value of the fraction ‘f’ variable. The red-highlighted field denotes the variable as a constraint to be added into the fitting error.

![Figure 4](image2.png)

**Figure 4**
Screenshot of the EXAFS fitting results page.
2.6. FEFF EXAFS fitting module

The IFEFFIT core engine provides a quick, robust fitting procedure for EXAFS. The SIXPACK GUI gives a simple form interface where the model is built in the traditional FEFFIT manner of terms and variables. Several templates are provided to assist the beginner in constructing simple shell-by-shell models. Owing to the unconstrained nature of IFEFFIT variable definitions, complex mathematical relationships can be constructed between fitted variables and EXAFS parameters. This leads to the ability to build models of complexity that are limited only by the imagination of the user. For example, models can include features such as fitting fractional components in the EXAFS, using bond angles as fitted parameters and interpolating between FEFF paths. Parameters and mathematical expressions in the fitting process can be ‘guessed’ (fully evaluated and updated at each iteration of the fit), ‘set’ (fixed value) or ‘defined’ (evaluated at the start of the fitting process, but not updated at each fitting iteration). Variables may be set as a restraint, where the variable value is added to the square error as part of the minimization routine. Restraints can be easily managed and given a variety of strengths to control the range or limits of parameters in the fit. In addition to the standard fitting above, EXAFS fits can be performed to a single or series of data $k$ weighting. Fitting schemes can be saved for later use and altered manually using any text editor.

The EXAFS fitting module of SIXPACK is displayed with the path-setup page in Fig. 2 and the variable-setup page in Fig. 3 to illustrate the GUI and several of the fitting features described above. The example shown is for sample data collected at the As K edge with a mixture of As–O and As–S first-coordination shells. The model incorporates these two shells, fitting their distances in real $R$ space and each shell with its own Debye–Waller factor. Since the coordination is a mixture of tetrahedral As–O and linear As–S, the coordination numbers for each shell are defined using the variable $f'$ as the fraction of As–O tetrahedral coordination, and $(1 - f')$ as the remainder of As–S coordination. Since $f'$ must be valid between values of 0 and 1, a variable ‘ferror’ is defined as a penalty for lying outside the nominal range and is used as a constraint in the fitting. The resultant fit values and their associated error bars are shown in the screenshot from SIXPACK in Fig. 4, and the data and fit are shown in Fig. 5.

3. Conclusions

SIXPACK is freely available online (https://www.sans-xrays.com/) and continues to be in development. Most forms of basic XAS and general fitting analyses are supported at this time. Support is provided for data formats of many synchrotron sources, including the XDI format. Comments and questions from users are welcomed. Future versions of SIXPACK (versions greater than 2.0) will migrate from IFEFFIT to usage of LARCH (Newville & Ravel, 2020) as the primary analysis engine, and include the ability to share fitting parameters in the linear combination fitting routines.

Acknowledgements

The author would like to thank Matt Newville for his effort in writing and supporting IFEFFIT, without which SIXPACK would have never started, and Bruce Ravel for providing much of the inspiration and the look and feel of many portions of the GUIs of SIXPACK. The many users of SIXPACK who have provided constant input and feedback into the direction of development of the software are also thanked.

References


