

Computer Programs

J. Synchrotron Rad. (2000), **7**, 283–286

***XAS-Collect*: a computer program for X-ray absorption spectroscopic data acquisition**

Martin J. George

Stanford Synchrotron Radiation Laboratory, Stanford Linear Accelerator Center, Stanford University, PO Box 4349, MS 69, Stanford, CA 94309, USA.

E-mail: mgeorge@ssrl.slac.stanford.edu

(Received 24 November 1999; accepted 4 May 2000)

A computer program for X-ray absorption spectroscopy (XAS) data acquisition, called *XAS-Collect*, is described. The software employs the X Windows system to provide a pull-down menu, dialog-box style of graphical user interface. The software provides state-of-the-art features while still being sufficiently simple for novice users.

Keywords: X-ray absorption spectroscopy; XAS; EXAFS; data acquisition.

1. Introduction

X-ray absorption spectroscopy (XAS), a relatively new spectroscopic technique originating in the early 1970s, has nevertheless in recent years experienced a rapid increase in use in a very wide variety of fields (Koningsberger & Prins, 1988). Approximately one-third of data collected at the Stanford Synchrotron Radiation Laboratory (SSRL) is XAS data, and SSRL's XAS users range from soil scientists, through structural molecular biologists, to chemists, materials scientists and solid-state physicists. Taking account of this wide range of user backgrounds, a flexible yet user-friendly standard for XAS data acquisition has been developed. The design goals of this program, called *XAS-Collect*, are to accommodate all of the sophisticated requirements of a state-of-the-art XAS experiment, but to be sufficiently simple to use to

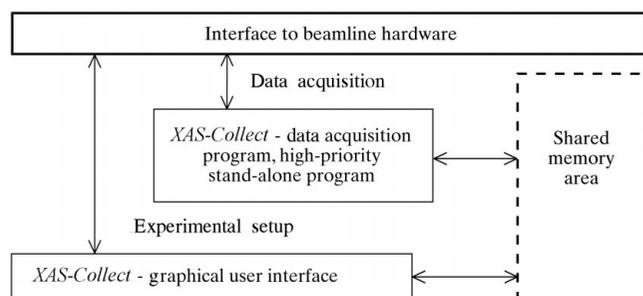


Figure 1
Schematic diagram showing the architecture of *XAS-Collect*.

allow novice users to make the most effective use of their beam time. To this end, the program incorporates automated features and has been carefully designed to facilitate a rapid experimental startup. To further assist in startup, the software has been designed to perform an XAS experiment in logical steps, starting with beamline alignment, then detector setup, file preparation, and initializing an experimental run. This paper describes the architecture and graphical interface of *XAS-Collect*.

2. Program architecture

Fig. 1 shows the basic architecture of the program. It is separated into two components: a data collector and a graphical user interface. The data collector is a compact very carefully written program with limited capabilities, which is designed to be extremely robust. The graphical user interface, on the other hand, is a large sophisticated multifunctional program which accepts user input, controls the collector, and performs such tasks as beamline alignment, data plotting and manipulation. The philosophy behind this design is that, should the user close the graphical user interface (or in the unlikely event of a program crash), the collector will remain active and there will be no interruption in data acquisition. The user interface can then be launched again and communication with the collector re-established. This type of architecture, in which different aspects of program function are divided between separate processes, has the advantage of conferring considerable stability because the operating system can prevent one process from interfering with another. This technique is known as storage protection, and is fundamental in modern programs and systems (*e.g.* Deitel, 1990). The collector and the graphical user interface communicate with each other using an area of shared memory, as shown in Fig. 1. In addition, other programs can access the experimental data, and the system also includes a simple interface (called *Spy*) for remote viewing of the current experiment.

3. The collector program

The collector is a separate program which is launched by the graphical user interface and subsequently runs independently at high priority until such time as the experiment is complete. The collector depends solely upon three files for its initial input. The first of these contains details of the experiment (*e.g.* output file name, comments *etc.*) plus the file names of the two other files,

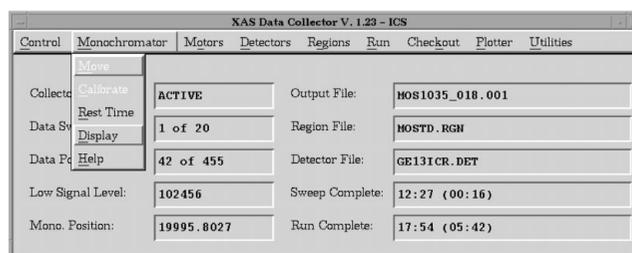


Figure 2
Main window of *XAS-Collect*.

which are called the ‘detectors file’ and the ‘regions file’. These latter two files contain details of the hardware to be accessed, and the motor motions (typically of a monochromator) required for the experiment. When the collector is active it can receive instructions from the graphical user interface (or from the *Spy* utility). These include pausing and stopping the experimental run, changing the number of scans (for a multiscan experiment) and restarting a paused experiment. Capabilities are included in the collector to suspend data acquisition should the X-ray beam dump, and to resume it when beam is restored.

4. The graphical user interface

Graphical user interfaces are now commonplace in data-acquisition software, and much effort has been put into the design of the *XAS-Collect* interface in order to make it intuitive and simple for the novice user. The main window of the *XAS-Collect* graphical user interface is shown in Fig. 2. The window includes areas providing information about the current experimental status, and pull-down menus to control various aspects of the experiment. Apart from some plotter functions (e.g. zoom), all operations can be controlled by keyboard short-cuts which are shown using an underscore; for example, alt-M opens the monochromator menu (shown in the figure).

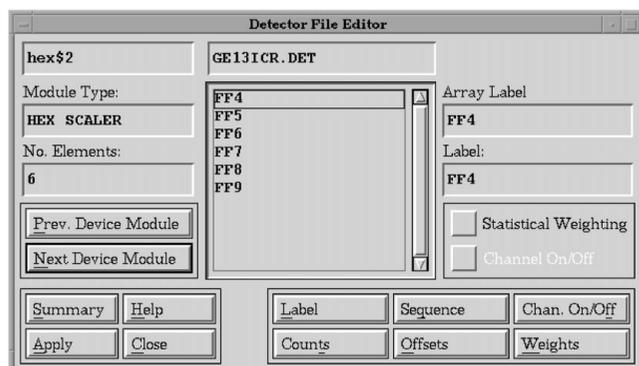


Figure 3
XAS-Collect detectors file dialog box.

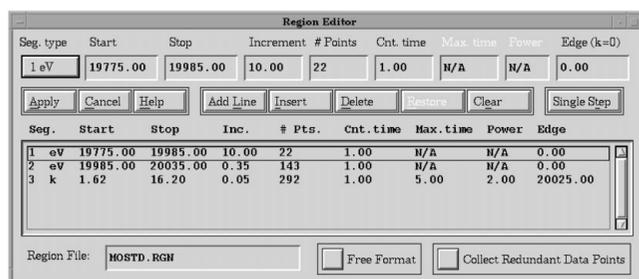


Figure 4
XAS-Collect region file editor box. The upper portion of the window contains editable fields while the scrolling list below displays the different segments which comprise the region.

4.1. Interface to beamline hardware: the detectors file

The interface to beamline hardware is defined within a ‘detectors file’, which contains a list of all physical devices to be used in data acquisition, together with details of each device and instructions for construction of the output data file. For our purposes we define ‘devices’ as any instruments which can be accessed by the computer. These might include motors, counters, real-time clocks, analog-to-digital converters, digital-to-analog converters and a variety of more specialized instruments. Details maintained on each device include a unique label for each data channel from the device, and an array position for the output file. Also included are device-specific details such as dark currents (e.g. for counters attached to detectors), and statistical weights for multi-element array detectors. Finally, the low-signal information for detecting X-ray beam loss is contained in the detectors file. Fig. 3 gives an example of a typical detectors file dialog box showing the labels for a counter device.

4.2. Motor motion control during data acquisition: the regions file

The motor, or combination of motors, which is most commonly associated with XAS data acquisition is the beamline monochromator. *XAS-Collect* is capable of scanning any computer-controlled motor or combination of motors, but for our purposes we will limit discussion to the monochromator. A region is defined as the combination of monochromator motions which will occur during data acquisition. A region contains different components, which are known as segments. Fig. 4 shows a regions dialog box containing three segments: a pre-edge segment with large steps in eV units, a near-edge segment with small steps, also in eV units, and an EXAFS segment with steps in reciprocal Ångström units (k) with count times weighted by k^2 , to provide better signal-to-noise ratios at the high- k end of the scan. Optional safeguards against non-contiguous and overlapping segments, and collection of redundant points (e.g. a monochromator move which would be less than one motor step), are built into the region editor. Regions containing reasonable segments can be automatically generated given the absorption edge and element to be investigated. Fig. 5 shows the automatic region generator dialog box. The options selected gave rise to the region shown in Fig. 4.

4.3. Alignment tools

As stated above, the graphical user interface is multifunctional. It provides facilities for simple alignment of beamline optical

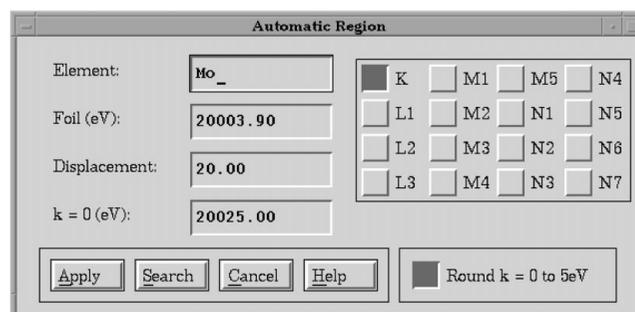


Figure 5
Automatic region file generator dialog box.

components. Fig. 6 shows the motor move dialog box and an example of a motor scan in which the vertical position of a protein crystal sample is scanned while monitoring the output of a single channel of a Ge fluorescence array detector. Any computer-controlled motorized component of a beamline can be scanned using the simple alignment tool.

4.4. Control of the experimental run

Once an appropriate detector and region file have been selected, an experimental run can be initiated. A run is typically comprised of a number of sweeps which give rise to individual data files. The run is started using a dialog box which accepts the file name, the number of sweeps, and descriptive comment lines which are stored in the data files. Prior to the run being initiated the program performs checks on the availability of the devices specified in the detectors file, on the validity of the regions files and the availability of X-ray beam.

Once a run is initiated, the graphical user interface relinquishes control of all beamline devices being used by the collector program. As mentioned above, the run can be paused, stopped or re-started, and the number of sweeps modified at any time. The data files produced by *XAS-Collect* are mirrors of the shared memory area, and thus are in binary format by default. A simple program to convert these files into a pure ASCII format is provided. The raw data files contain an ASCII header which lists the regions and detector files used, data column labels, electronic dark-current levels for each device, comments entered by the user, plus various other experimental details. The binary data area follows the ASCII header. The collector automatically subtracts the electronic dark currents from the data; the dark-current information for each device is also stored in the data file.

4.5. XAS plotter

Current data or old data from files can be displayed using the built-in plotter. Individual or sums (optionally weighted) and ratios of the array positions in the output file can be displayed as abscissa or ordinate. A variety of line styles, data-point styles and colors can be selected, and automated and manual scaling of plots

is available. Details of the plots can be saved, so that similar plots can be generated instantly, and useful default plots are made available with the standard distribution. Capabilities to automatically overlay plots and to zoom in on features of interest are also included.

Basic data-reduction capabilities are incorporated. Data can be averaged, the pre-edge and spline can be subtracted to generate the extended X-ray absorption fine structure (EXAFS), and an EXAFS Fourier transform may be calculated. Other features such as computation of derivatives are available, which can be useful for monochromator calibration. These on-line analysis features are intended to give the user a first look at the data to allow judgments to be made about the direction of the experiment. These on-line analysis features are complemented by more extensive analysis code available on-line at SSRL (*e.g.* *EXAFS-PAK*; George, 1996). Finally, all graphical output produced by the plotter can be printed or saved to a postscript file.

4.6. Other features

The code contains a large number of additional features which will not be described here because of space limitations. These include automatic generation of log files describing data-acquisition activities, and an extensive printing utility for generating summaries of region files, detector files *etc.* A menu-driven comprehensive help facility is also available, which incorporates two levels of help, with brief and detailed descriptions. Finally, full documentation is available both on-line and in printed form (Pickering, 1995).

5. Program portability and availability

The programs are written entirely in ANSI standard C, with very little operating-system-dependent code. The primary motive behind this design is to maximize the portability of the code. While the initial implementation of the program is under OpenVMS, the X Windows toolkit and graphical user interface compile and operate under other operating systems which provide X Windows systems, such as UNIX, Linux and Windows NT (using third-party X Windows development libraries and server).

Access to beamline hardware, such as motors and counters, is accomplished using a separate software layer. While the current implementation is dependent on this layer, the *XAS-Collect* code is written in a highly modular manner. For example, all motor movement is accomplished using a single subroutine. Moving the code to another platform without the underlying control layer would involve only a small number of modifications.

The *XAS-Collect* software is installed on the majority of beamline computers at SSRL and is now being considered at other institutions. Interested parties should contact the author.

6. Summary

XAS-Collect has now been in routine use at SSRL for five years. It has been well received by the user community and has benefited from their input. It has been used to collect the data giving rise to more than 400 publications in the primary scientific literature. We believe that most or all of the initial design goals have been achieved, although the program continues to be expanded and developed.

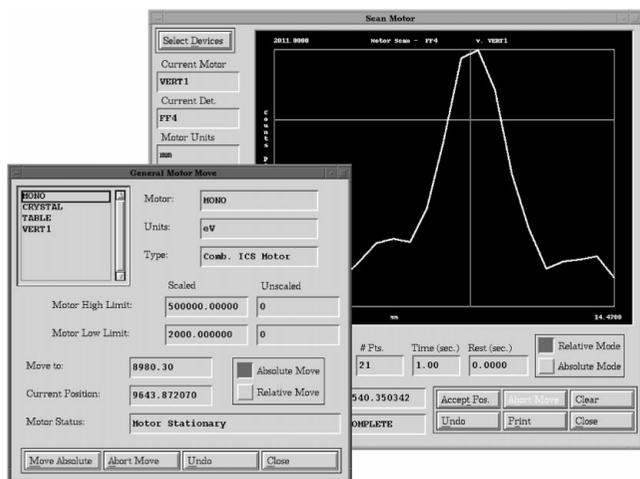


Figure 6
Motor move and motor scan dialog boxes.

The Stanford Synchrotron Radiation Laboratory is funded by the Department of Energy, Office of Basic Energy Sciences. The Structural Molecular Biology Program is supported by the National Institutes of Health, National Center for Research Resources, Biomedical Technology Program, Division of Research Resources. Further support is provided by the Department of Energy, Office of Biological and Environmental Research.

References

- Deitel, H. M. (1990). *An Introduction to Operating Systems*. Reading, MA: Addison-Wesley.
- George, G. N. (1996). *EXAFSPAK*, <http://ssrl.slac.stanford.edu/exafspak.html>.
- Koningsberger, D. C. & Prins, R. (1988). Editors. *X-ray Absorption. Principles, Applications, Techniques of EXAFS, SEXAFS and XANES*. New York: John Wiley and Sons.
- Pickering, I. J. (1995). *XAS-Collect Users' Manual*, <http://www-ssrl.slac.stanford.edu/~xas/xasman.html>.