SinSPECt Documentation

Release 0.6

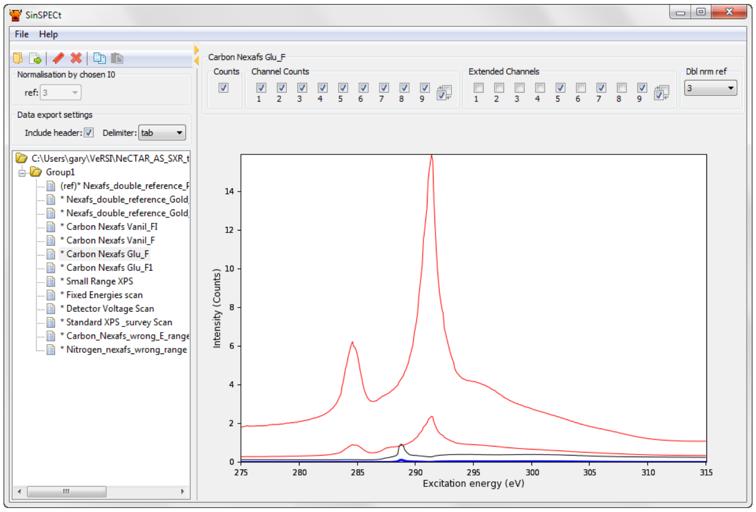
Gary Ruben

September 06, 2016

Contents

SinSPECt [1] (Soft x-ray spectrum inSPECTor) is a data explorer for Soft X-Ray spectroscopy data stored in the SPECS XML format.

- Reads SPECS XML format files saved from SpecsLab2.
- Graphical exploration of data regions.
- Exports columnar ASCII with choice of delimiter and optional headers.
- Supports normalisation of data to internally available data channels.
- Cross-platform. Runs on Windows/Mac OSX/Linux
- Released under the Modified BSD-license



Normalisation by chosen I0

ref: 3

Data export settings

Include header: ☑ Delimiter: tab

C:\Users\gary\VeRSI\NeCTAR_AS_SXR_t
Group1
(ref)* Nexafs_double_reference_F
* Nexafs_double_reference_Gold
* Nexafs_double_reference_Gold
* Carbon Nexafs Vanil_FI
* Carbon Nexafs Vanil_F
* Carbon Nexafs Glu_F
* Carbon Nexafs Glu_F1
* Small Range XPS
* Fixed Energies scan
* Detector Voltage Scan
* Standard XPS _survey Scan
* Carbon_Nexafs_wrong_E_range
* Nitrogen_nexafs_wrong_range

Carbon Nexafs Glu_F

Counts: ☑

Channel Counts: ☑ 1 ☑ 2 ☑ 3 ☑ 4 ☑ 5 ☑ 6 ☑ 7 ☑ 8 ☑ 9

Extended Channels: □ 1 □ 2 □ 3 □ 4 ☑ 5 □ 6 ☑ 7 □ 8 ☑ 9

Dbl nrm ref: 3

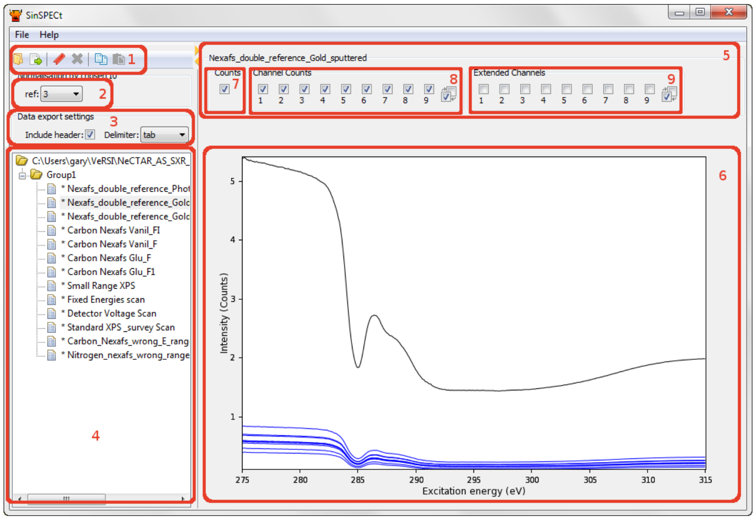
[1] SinSPECt.

Contents

1.1 Usage

This page describes the usage of SinSPECt.

The SinSPECt window starts with two main areas, divided by a movable vertical splitter bar. After loading a file and selecting a region, the window should look similar to that in the figure.



ref: 3

Include header: ☑ Delimiter: tab

Counts: ☑

Channel Counts: ☑ 1 ☑ 2 ☑ 3 ☑ 4 ☑ 5 ☑ 6 ☑ 7 ☑ 8 ☑ 9

Extended Channels: □ 1 □ 2 □ 3 □ 4 □ 5 □ 6 □ 7 □ 8 □ 9

Fig. 1.1: SinSPECt screenshot (Windows version) identifying parts of the user interface.

The parts of the user interface are: 1. Toolbar 2. Single normalisation extended channel choice 3. Exported ASCII file format settings 4. Tree editor reflecting SPECS .xml file structure 5. Region selector panel 6. Region plot window 7. Counts (export region) checkbox 8. Channeltron channel checkboxes 9. Extended channel checkboxes

1. **Toolbar.** From left-to-right, the toolbar buttons allow you to 1a) Open a SPECS file, 1b) Export all regions marked for export, 1c) Set the region currently selected in the tree editor as the double-normalisation reference, 1d) Clear any double-normalisation reference and disable double-normalisation, 1e) Copy the Checkbox states of the currently selected region to a clipboard, 1f) Paste the clipboard state to the current tree editor selection.

   1a) Open SPECS file.

   SPECS files are organised into a 2-level hierarchy, with the possibility of multiple Groups and the possibility of multiple Regions within each group.

Note: Name mangling on import

SPECS files can potentially contain many Groups with duplicated identical naming. Within these, Regions could also be named identically. To avoid any problems that this would cause, SinSPECt alters the names to ensure uniqueness by appending sequential numbers to any duplicate. e.g. If a group contained several regions named 'a', 'b', 'c', 'a', 'b', 'a', SinSPECt would change these to 'a', 'b', 'c', 'a-1', 'b-1', 'a-2' and write files named 'a.xy', 'b.xy', 'c.xy', 'a-1.xy', 'b-1.xy', 'a-2.xy'.

1b) Export marked regions.

SinSPECt exports .xy files as columnar ASCII in files with names matching the Region names. The files are written into directories matching the Group names. Regions marked with a '*' or '+' will be exported. These correspond to those Regions whose Counts checkbox (box 7 in the figure) is checked. Initially, all regions are set to be exported, as indicated by a '*' marker.

Note: A '+' marker indicates that the displayed and exported Counts column data is formed from a subset of the possible Channeltron Channel Counts summands. Only if all checkboxes in box 8 in the figure are checked will this marker be a '*'.

Warning: Beware of a bug in the export dialog box which affects the MS Windows version. Clicking the "Export marked regions" toolbar button opens a file browser with a button allowing you to make a new folder. After creating a folder using this button, if you rename the folder to something other than the default name "New folder," the folder is not selected correctly. The symptom of this bug is that the text field below the tree view is not updated to the new folder. You will need to click another folder in the tree view then click on the newly created folder to update the text field before clicking "OK" to export your data.

1c) Set region as double normalisation reference.

Clicking the bookmark toolbar button enables double normalisation and sets the currently selected region as the reference region, indicated by the text (ref) in the label alongside the reference region. When double normalisation is enabled, drop-down selectors appear to the right of the selection panel checkboxes. See the Double Normalisation section in *Processing* for further details.

2. **Single normalisation extended channel choice.** SinSPECt operates in one of three modes: 1. No processing, 2. Single normalisation, 3. Double normalisation. SinSPECt starts in the No processing mode, indicated by this selector being set to "None". Selecting any value other than None changes the operation mode to Single normalisation and establishes the Extended Channel corresponding to the displayed number as the Single normalisation reference. Setting this to "None" returns SinSPECt to the No processing mode. The selector is disabled when operating in Double normalisation mode.
3. **Exported ASCII file format settings.** SinSPECt exports .xy files as columnar ASCII. These settings allow header columns to be disabled and the delimiter to be set.
4. **Tree editor reflecting SPECS .xml file structure.** When a SPECS file is loaded, its structure is reflected in the navigation tree structure displayed in this panel. Files may contain many Groups, each of which may contain many Regions. Mouse and keyboard actions applying to tree editor selections are described in the menu item *Help → About...* window.
5. **Region selector panel.** This panel contains any Region-related settings. Its state updates according to the Region selected in the tree editor.
6. **to 9. Region plot window and Region selector panel checkboxes.** The plot window contains plots corresponding to the checkbox states in the Region selector panel. The mouse allows zooming and panning in this panel (described in the menu item *Help → About...* window). Channeltron Channel Counts are

shown in blue, their Counts sum is shown in black, and Extended Channels are shown in red. Plots are normalised according to the description in the *Processing* section. See above note in (1b). The Channel Counts (box 8) and Extended Channels (box 9) checkbox groups also contain buttons that allow quick checking and unchecking of all checkboxes within the checkbox group.

1.2 Processing

This page describes the processing that SinSPECt performs.

1.2.1 Exported data

Regions selected for export are saved as .xy files. .xy files are ASCII files containing two optional header lines followed by columnar numerical data. Fields are TAB separated by default, allowing the file to be imported into MS Excel easily, e.g. by drag-and-drop. Both header lines start with the # comment character. The first line contains text representations of some region-specific metadata fields and indicates whether the data is single- or double-normalised, as described below. An example is:

```
#"Analyzer mode:FixedAnalyzerTransmission, Dwell time:0.1, Pass energy:5.0, Lens mode:MediumArea:1.5
```

The second line contains column headings for the numeric data. An example is:

```
#"Binding Axis" "Counts 1+2+3+4+5+6+7+8+9" "Channel 1 counts" "Channel 2 counts" "Channel 3 counts"
```

An example of a row of numeric data is:

```
110 711 116 91 157 36 74 85 102 41 9 0 21298 75412 0 11253 0 0 56360 0
```

which contains the following from left-to-right: x-value, counts, 9-channeltron-channels, 9-extended-channels. Occasionally regions in the .xml file may contain no channel data or no extended channel data. In these cases, the data is zero-filled with the correct number of entries to match the number of x-values. By default, exported region data reflects the value read from the .xml file exactly. The counts value is determined by summing those channeltron channels with corresponding checked checkboxes. Unchecking a checkbox removes that channel from the sum. The contributing channels are reflected by the header of second column; e.g., “Counts 1+2+3+4+5+6+7+8+9” above indicates all channels were summed to get the values in this column. Data may also be normalised as described below under normalisation and double normalisation.

Note: SinSPECt exports raw Channeltron Channel Counts, not counts per second. If you want to convert to counts per second in your analysis, you will need to divide exported counts values by the Dwell time.

1.2.2 Data normalisation

SinSPECt operates in one of three modes: 1. No processing, 2. Single normalisation, 3. Double normalisation.

No processing

In this mode all displayed plots and exported data correspond to the raw channeltron and extended-channel data.

Note: Enabling No processing mode

This mode is enabled when the drop-down selector in the “Normalisation by chosen I0” GUI group is disabled or set to “None” and when the double-normalisation mode is disabled, indicated by the toolbar ‘X’ button being greyed-out.

Single Normalisation

In Single-normalisation mode, all displayed and exported data is divided by the data in the selected extended channel $\mathbf{e}_r$ (the data in the selected extended channel is left unnormalised).

Note: Enabling Single Normalisation mode

The “Normalisation by chosen I0” GUI group contains a drop-down selector labelled “ref:” which selects the extended channel that all y-data values will be divided by. The selector is enabled whenever an .xml SPECS data file is loaded and whenever the double-normalisation mode is not enabled. If the selector is set to “None”, all normalisation is disabled: plots and exported data will correspond to the raw data. If the selector is set to a value 1-9, normalisation is enabled.

Warning: It is possible to set the reference channel $\mathbf{e}_r$ to an extended channel that may contain zeros in some regions, resulting in divide-by-zero errors. In this case, the plot window will be blank and errors will be produced for exported regions that cause errors. If any errors occurred during export, a notification is displayed, the filenames of affected .xy data files are prepended with the text *ERRORS_*, and a header line indicating the error is written to the file.

During preview and export, data is processed as follows.

$$\mathbf{C} = \sum_i a_i \mathbf{c}_i,$$

where $\mathbf{C}$ is the Counts vector, $i = 1..9$ is an index over the channels, $\mathbf{c}_i$ is the vector of channel data, a_i is 1 if the i th channel is enabled or 0 otherwise. The channels $\mathbf{c}_i$ and extended channels $\mathbf{e}_i$ are those values read from the .xml SPECS file. The normalised Counts $\mathbf{C}'$ are then

$$\mathbf{C}' = \mathbf{C}/\mathbf{e}_r,$$

where $\mathbf{e}_r$ is the specified reference extended channel. In addition to the Counts, the single-normalised channel counts $\mathbf{c}'_i$ and extended channel counts $\mathbf{e}'_i$ are determined according to

$$\mathbf{c}'_i = \mathbf{c}_i/\mathbf{e}_r,$$

and

$$\mathbf{e}'_i = \mathbf{e}_i/\mathbf{e}_r, i \neq r.$$

Here, the condition $i \neq r$ shows that we choose not to normalise the reference extended counts channel data to be identically 1, i.e. the exported values are simply

$$\mathbf{e}'_i = \mathbf{e}_i, i = r.$$

This enables the normalisation procedure to be undone if desired, with access to only the exported data file.

Double Normalisation

In Double-normalisation mode, all displayed and exported data is divided first by the extended channel specified for that region then further by the ratio of the two channels specified in a reference region (the data in the selected extended channel is left unnormalised).

Note: Enabling Double Normalisation mode

Clicking the bookmark toolbar button enables double normalisation and sets the currently selected region as the reference region R . The text *(ref)* appears in the label alongside the reference region in the tree editor to indicate this. Clicking the 'X' button adjacent to the bookmark clears double normalisation mode. When double normalisation is enabled, drop-down selectors appear to the right of the selection panel checkboxes. These enable setting of the extended channels used to compute the double-normalised results. For all regions other than the reference region, the group contains one drop-down selector. This allows selection of the extended channel $\mathbf{e}_r$ (see description below). For the reference region, the selector panel contains two drop-down selectors that allow setting of the values s and $\mathbf{e}_r^R$ (see description below).

Warning: It is possible to set the reference extended channel in the current reion $\mathbf{e}_r$ or that of the reference region $\mathbf{e}_r^R$ to a channel that may contain zeros in some regions, or whose x-axis ranges differ. In both cases, the plot window will be blank and errors will be produced for exported regions that cause these error types. If any errors occurred during export, a notification is displayed, the filenames of affected .xy data files are prepended with the text *ERRORS_*, and a header line indicating the error is written to the file.

During preview and export, the double normalised Counts $\mathbf{C}''$ is

$$\mathbf{C}'' = \sum_i a_i \mathbf{c}_i / \mathbf{e}_r / (M^R / \mathbf{e}_r^R),$$

where M^R depends on the drop-down menu selection $s \in \{\text{Counts}, 1..9\}$ as follows.

$$M^R = \mathbf{e}_s^R, \text{ if } s \in 1..9,$$

or, if $s = \text{Counts}$

$$M^R = \mathbf{C}^R = \sum_i a_i^R \mathbf{c}_i^R, \text{ if } s = \text{Counts}.$$

Here $\mathbf{e}_r$ is the reference extended channel in the current region, $\mathbf{e}_r^R$ is the reference extended channel in the reference region R. In addition to the Counts, the double-normalised channel counts $\mathbf{c}_i''$ and extended channel counts $\mathbf{e}_i''$ are determined according to

$$\mathbf{c}_i'' = \mathbf{c}_i / \mathbf{e}_r / (M^R / \mathbf{e}_r^R)$$

and

$$\mathbf{e}_i'' = \mathbf{e}_i / \mathbf{e}_r / (M^R / \mathbf{e}_r^R).$$

As for the single normalisation case, we choose not to normalise the reference extended counts channel data to be identically 1, i.e. the exported values are simply

$$\mathbf{e}_i'' = \mathbf{e}_i, i = r.$$

In order to enable the normalisation procedure to be undone if desired, a column is appended that contains the $M^R/\mathbf{e}_r^R$ values, allowing reversal of the processing with access to only the exported data file.

1.3 Installation

SinSPECt is written in Python (2.7 at the time of writing) and has several module dependencies that require installation. For Windows, Linux and Mac users the easiest way to install these dependencies is to install one of the Enthought

Python Distributions. The EPDFree distribution contains all the modules on which SinSPECt depends. A typical user can just download and install EPDFree then download and unpack SinSPECt and run it immediately.

Some users might prefer alternatives to EPDFree. For example Academic or Commercial users may prefer to use one of Enthought's other distributions. Other popular Python distributions include Python(x,y), WinPython and ActiveState's ActivePython, all of which will require installation of at least some of the following dependencies:

1.3.1 Python package dependencies

SinSPECt depends on the following Python modules being installed in the Python environment

- numpy
- traits [part of the Enthought Tool Suite (ETS) package]
- traitsui (part of ETS)
- chaco (part of ETS)
- pyface (part of ETS)
- wxPython

Those wishing to build the documentation from source will also need Sphinx. For those familiar with installing Python packages, the dependencies can be found in the central repository of Python modules. For Microsoft Windows users, Christoph Gohlke (C.G.) maintains a useful repository of Windows installers for many modules. Linux users can typically find the dependencies using their package manager (e.g. synaptic or yum). Mac users should visit the individual sites linked above for instructions.

1.3.2 Example 1. Windows installation using Enthought Python Distribution (recommended)

1. Visit the SinSPECt download page and follow the instructions to obtain the EPD Free Python installer and the SinSPECt installer files for Windows. You may wish to visit the Enthought website, directly, and choose one of Enthought's other Python distributions. Note, EPD Free satisfies the dependencies listed above.
2. Run the epd_free-*.msi installer to install EPD Free.
3. Verify that Python is running correctly. e.g. for Windows 7, click on `Start Menu|All Programs|Accessories|Command Prompt`. At the > prompt type `python -c "print 'hello world'"` noting single and double quotes. Verify that `hello world` is displayed.
4. Running the SinSPECt installer will install SinSPECt and create a start menu entry.

1.3.3 Example 2. Windows installation using Python(x,y) (optional)

1. Visit the Python(x,y) downloads page and install a distribution.
2. Verify that Python is running correctly (see *Example 1. Windows installation using Enthought Python Distribution (recommended)*).
3. Visit the SinSPECt download page and follow the instructions to obtain the SinSPECt setup file.
4. Running the installer will install SinSPECt and create a start menu entry.

1.3.4 Example 3. Windows installation on a system with Python already installed (experienced)

Note: untested.

1. Check the *package dependency list* above.
2. The easiest way here is to use the packages provided by Python(x,y) and/or Christoph Gohlke. Install the required dependency, taking care to choose packages for Python 2.7 and to choose the 32 or 64 bit package version that matches your Python version. Install in turn `numpy` (if in doubt choose the numpy-MKL-... version), `scipy`, `matplotlib`, `wxPython`, and finally `ETS`.

1.3.5 Example 4. Linux installation using Enthought Python Distribution (recommended)

1. Visit the SinSPECt download page and follow the instructions to obtain the EPD Free Python installer and the SinSPECt .zip package for Linux. You may wish to visit the Enthought website, directly, and choose one of Enthought's other Python distributions. Note, EPD Free satisfies the dependencies listed above.
2. Run the epd_free-*.sh shell script to install EPD Free.
3. Verify that Python is running correctly. e.g. for Ubuntu, open a terminal. At the `$` prompt type `python -c "print 'hello world'"` noting single and double quotes. Verify that `hello world` is displayed.
4. The main SinSPECt application file is `app.py` in the directory into which SinSPECt was unpacked.
5. SinSPECt can be started by running `./start_sinspect.sh` or `./app.py`. To do this, at the `$` prompt type `chmod 777 start_sinspect.sh app.py` followed by, for example, `./start_sinspect.sh`

1.3.6 Example 5. Linux installation using synaptic (experienced)

Note: untested.

This description is for Ubuntu Linux. yum packaged names in Fedora Linux flavours should have similar names.

1. First, verify that Python2.7 is running correctly. e.g. for Ubuntu, open a terminal. At the `$` prompt type `python -c "import sys; print sys.version"`. Verify that a string displays identifying a 2.7 branch version of Python.
2. Using synaptic or `apt-get install <package>` install the following packages: `python-numpy`, `python-scipy`, `python-matplotlib`, `python-traits`, `python-traitsui`, `python-chaco`, `python-pyface`, `python-wxgtk2.8`
3. Visit the SinSPECt download page and follow the instructions to obtain the package.
4. The main application file is `app.py` in the directory into which SinSPECt was unpacked.
5. SinSPECt can be started by running `./start_sinspect.sh` or `./app.py`. To do this, at the `$` prompt type `chmod 777 start_sinspect.sh app.py` followed by, for example, `./start_sinspect.sh`

1.3.7 Example 6. Mac OSX installation (recommended)

1. Visit the SinSPECt download page and follow the instructions to obtain the EPD Free Python installer and the SinSPECt .zip package for Mac OSX. You may wish to visit the Enthought website, directly, and choose one of Enthought's other Python distributions. Note, EPD Free satisfies the dependencies listed above.
2. Run the epd_free-*.dmg installer to install EPD Free.

3. Move the .zip package to the Applications folder.
4. Double click the Application .zip package to unpack it the first time.
5. Now you can double click the package to start SinSPECt.

1.4 About SinSPECt

Please send bug reports and suggestions to *sinspect _at_ synchrotron.org.au*

Software authors:
Gary Ruben *gruben _at_ versi.edu.au*, Victorian eResearch Strategic Initiative (VeRSI)
Kane O'Donnell *kane _dot_ odonnell _at_ synchrotron.org.au*, Australian Synchrotron

Software home:
http://www.synchrotron.org.au/sinspect
Software source:
http://github.com/AustralianSynchrotron/sinspect

Recognition of NeCTAR funding: The Australian Synchrotron is proud to be in partnership with the National eResearch Collaboration Tools and Resources (NeCTAR) project to develop eResearch Tools for the synchrotron research community. This will enable our scientific users to have instant access to the results of data during the course of their experiment which will facilitate better decision making and also provide the opportunity for ongoing data analysis via remote access.

CHAPTER 2

Indices and tables

- genindex
- modindex
- search