fcc_analyzer_PyQt4
a GUI to visualize *FCclasses* output

Javier Cerezo
j.cerezo@pi.iccom.cnr.it

VERSION: 0.1
June 1, 2016

1 Description

fcc_analyzer_PyQt4 is a graphical user interface (GUI) written in python and relying on the PyQt4 and matplotlib libraries. It reads the output from *FCclasses* (for a TI calculation), namely fort.21 (assignments) and fort.22 (spectral histogram), and opens an interactive plot that facilitates the analysis of the transitions, the investigation of different broadening schemes and the comparison with reference spectral data.

2 Installation

2.1 Precompiled binary

A binary file generated with pyinstaller is provided. It is compatible with 64-bit Linux systems. The GLIBC version ≥2.17 is required. The binary and libraries are in the bin/ folder. The main program is bin/fcc_analyzer_PyQt4.

2.2 Python script

The source can be run directly as a python script, located in src/. python version 2.7 is used. The following modules are required (the minimum version tested in specified):

- matplotlib (version ≥ 1.4.2)
- numpy (version ≥ 1.9.1)
- PyQt4 (version ≥ 4.10.4)

3 Running the application

3.1 Launch the application

The application need to be invoked in the folder were the fort.21 and fort.22 files that will be analysed reside, with the following syntax:

```
fcc_analyzer_PyQt4 [flags]
```

The following optional flags are possible:

`-type (abs|emi|ecd|cpl)`	Type of $\mathcal{FC}$*classes* calculation performed.
`-maxC N`	Maximum class to by loaded (`N`≤ 7).
`-h`	Show help.

3.2 Using the application

The GUI consists of the following parts:

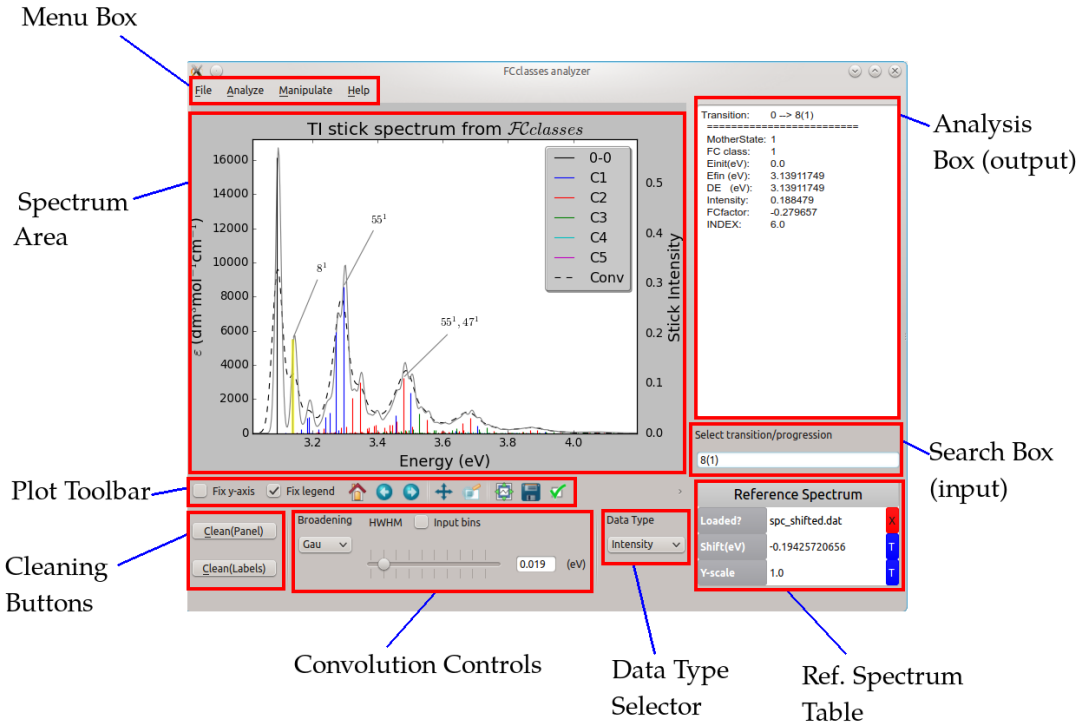

Figure 1: Screenshot of `fcc_analyzer_PyQt4` highlighting the different parts.

The interaction and analysis of the simulated spectrum are performed with the different widgets in the plot. Below, the different parts are described.

3.2.1 Menu Box

- File
  - Save plot: save plot to a png file.
  - Export to xmgrace: export plot data (including labels) to xmgrace file. A export assistant is first raised to select how to organize the plots:
    - Overlaid graphs: place the stick and spectra in different graphs, that are overlaid, as in the matplotlib plot.
    - Same graph: include sticks and spectra in the same graph. The spectra are scaled to normalized the maximum intensity.
  - Import plot: Load reference spectrum to the Spectrum Area. Units: (eV|cm^{-1}|nm).
  - Quit: exit the application
- Analyse
  - Momenta: compute momenta of the convoluted and write to the Analysis Box.
- Manipulate
  - Shift to simulated: shift reference spectrum (if loaded) to match the convoluted spectrum.
  - Scale to convoluted: scale reference spectrum (if loaded) to match the convoluted spectrum.
- Help
  - Instructions: show a short guide to use the application.
  - About: show general info about the application.

3.2.2 Spectrum Area

This area is interactive. The following actions are possible:

- Right-mouse click on a stick: highlight and get info about the transition (written into Analysis Box).
- Left-mouse click on a stick: add label over the stick
- Right-mouse click on a label: drag the label
- Right-mouse click on a legend line: activate/deactivate plot
- Press "+"/"-" keys to browse back and forward between highlighted stick

3.2.3 Analysis Box

In this box, the information about the highlighted transitions is shown, as well as the analysis of the momenta. This box is not editable, but the info can be copied.

3.2.4 Search Box

Selector of transitions. The general syntax is the following:

```
Mode1'(Quanta1'),Mode2'(Quanta2')...  --> Mode1(Quanta1),Mode2(Quanta2)...
```

where `Mode1'(Quanta1')` refer to mode `Mode1'` in the initial state that is excited `Quanta1'` quanta, and `Mode1(Quanta1)` are the equivalent entities for the final state.

The ground vibrational state is represented by a zero (`0`). When the initial state is in the ground state, the initial part, `0-->` can be omitted. Some examples:

- `8(1),9(2)`: select the transition from the ground initial state to the final state where mode 8 is excited 1 quantum and mode 9 is excited 2 quanta.
- `0-->8(1),9(2)`: same as above
- `8(1)-->0`: select transition from initial state excited 1 quantum on mode 8 to the final ground state.
- `0-->0`: select 0-0 transition

To select a progression in the final state, use `P` instead of the number of quanta. This keyword can only be used for one mode in the same selection. Some examples:

- `8(p)`: select progression of mode 8 in the final state.
- `0-->8(p),9(2)`: select progression of mode 8 in the final state when, simultaneously, mode 9 is excited 2 quanta.
- `1(1)-->8(p)`: select progression of mode 8 in the final state, starting from the initial state where mode 1 is excited one quantum.

3.2.5 Plot Toolbar

This is formed by two check boxes and the standard matplotlib toolbar with PyQt4 back-end. The check boxes are:

- Fix y-axis: whether or not scale y-axis when the convolution or the data type are updated.
- Fix legend: whether the legend can be displaced with the mouse of not.

The matplotlib toolbar include:

-  : zoom into (left mouse click) or out (right mouse click) a rectangle.
-  : move the spectrum (left mouse click) or zoom (right mouse click).
-  : export image.
-  : customize axes properties.

3.2.6 Cleaning Buttons

- Clean(Panel): remove highlight on transitions and clear the analysis box.
- Clean(Labels): clear all labels added to the spectrum

3.2.7 Convolution Controls

- Broadening selector [Gau|Lor]: type of broadening function
- HWHM slider and box: set the HWHM of the broadening function
- Input bins check box: use the bins given in the input fort.22 file. Otherwise, a new histogram with only 1000 bins is used (which is faster).

3.2.8 Data Type Selector

Select the type of spectrum: [Intensity|Lineshape].

3.2.9 Ref. Spectrum Table

Information about the reference spectrum (loaded with File->Import plot).

- Loaded?: whether reference spectrum is loaded. If so the file name is shown. The red button with the cross deletes the reference spectrum.
- Shift(eV): shift applied to the reference spectrum (in eV). This box is editable. The value always refer to the originally loaded value, unless the blue T button is pressed, which reset the reference value to the current one.
- Y-scale: scaling applied to the reference spectrum Y-axis. This box is editable. The value always refer to the originally loaded value, unless the blue T button is pressed, which reset the reference value to the current one.