pybranch.py : A python script to obtain transition probabilities from Fourier transform spectra

Gillian Nave, Jacob Ward
National Institute of Standards and Technology, Gaithersburg, MD

January 10, 2025

1 Description of program pybranch.py

This python program is designed to assist with the measurement of branching fractions in atomic spectra, particularly those measured using Fourier transform spectroscopy. It requires the following python libraries to be installed:

- glob
- os
- numpy
- matplotlib
- scipy
- tkinter

In addition to the main program, pybranch.py, it uses the following programs:

branchgrep: Used to find all of the lines in the input files that originate from a level

branchrc: Used to set initial values for the entry fields, the identified lines file, the levels file, the calculations file, and the extension for the files containing the branching ratios. Also sets the default values for the wavenumber discriminator to match the observations with the calculations, and the size of the plotting window.

nso_to_hdf: Includes routines to read the Xgremlin header, .dat, and .lin files.

2 Input files

All files use a key to identify the energy levels being used. This key must be common to all the files used, although other energy level descriptions may be used elsewhere in the file. This enables the program to match the same energy level from different sources, even when the value of the level may be different (e.g. values from older calculations, compared with modern measurements)

Identified lines file: A file of previously-identified lines, for example from a previous analysis of the spectrum or from the NIST Atomic Spectra Database, separated by column. This is printed at the top of the results window, and is meant to help the user spot any lines missing in the files of branching ratios, or any lines that may be problematic because they are blended. The format is:

Column 1: Wavenumber of transition.

Column 2: Lower energy level value in cm^{-1}.

Column 3: Upper energy level value in cm^{-1}.

Column 4: Lower configuration key - this must match the format in the energy levels and lifetimes file. Can't contain spaces.

Column 6: Upper configuration key - this must match the format in the energy levels and lifetimes file. Can't contain spaces.

Columns 7 - end: Optional notes (e.g. intensities, lists of blending transitions etc.). This can include spaces, and up to 54 characters are currently taken from the input file and printed in the log file.

An example is:

```
 8503.122    36101.52   84604.64      d5____a2G3 -     1F4p__u2F2*       21
48491.0570       0.00   48491.06      d5____a6S2 -     5D4p__z6P2*     2500
48474.553    20517.79   68992.35      d5____a4G3 -     3G4p__y4H4*       28
48473.07     20519.27   68992.35      d5____a4G4 -     3G4p__y4H4*        4
48451.940    25033.67   73485.61      d5____b4D3 -     3D4p__w4D3*       69
```

Lifetimes file: File of energy levels and lifetimes. The format is:

Column 1: Energy level - any format but can't contain spaces.

Column 2: J value.

Column 3: Energy level value in cm^{-1}.

Column 4: Parity - 0 or 1.

Column 5: Lifetime in ns. If none, should be '-'.

Column 6: Energy level key. This must match the format in the identified lines file.

An example is:

```
  5D4p__z6F4* 4.5     47464.55            0 4.2 5D4p__z6F4
  5D4p__z6F5* 5.5     47751.62            0 4.0 5D4p__z6F5
  d5____d2D2  2.5     47354.44            1 - d5____d2D2
  d5____d2D1  1.5     47372.53            1 - d5____d2D1
```

Calculations file: File of calculations to be used to calculate the residual and to compare the experimental results with theory. The format is:

Column 1: Wavenumber. Does not need to be an exact match to the wavenumbers in the spectra or identified lines file but should match within the discriminator specified in the branchrc.py file.

Column 2: The calculated transition probability in units of 10^6 s^{-1}. This is used for comparison with the measured transition probabilities and also for calculating the residuals.

Last column: The energy level key. This must match the format in the identified lines file and the lifetimes file.

An example is:

```
45246.3400    0.3883     2209.435    -2.94411  1.5   21824.11* 1|4P3)      1.5   67070.45* a3F4p_y4F1*
45246.6100    4.1274     2209.422    -1.91757   .5   21823.84* 1|4P3)      1.5   67070.45* a3F4p_y4F1*
45247.9300    4.0368     2209.357    -1.92724  2.5   21822.52* 1|4P3)      1.5   67070.45* a3F4p_y4F1*
45358.6800    0.0220     2203.962    -4.01645  2.5   19797.88* 2|5D4)4D    2.5   65156.56* a3F4p_z4G2*
45359.9400    0.2962     2203.901    -2.66596  3.5   20024.01* 2|5D4)4D    4.5   65383.95* a3F4p_z4G4*
45459.0000    0.1568     2199.098    -3.04100  2.5   19797.88* 2|5D4)4D    3.5   65256.88* a3F4p_z4G3*
```

Branching ratio files: Set of files of measured relative line intensities in units of photons per second. The filenames and extensions should match the pattern specified in 'all_spectrum_files' variable in the branchrc.py file. The first line has three numbers:

- The resolution of the spectrum in cm^{-1}.
- The lower wavenumber limit of the calibration in cm^{-1}
- The upper wavenumber limit of the calibration in cm^{-1}.

These three numbers are used to calculate the uncertainties of the branching fractions and transition probabilities.

The following lines contain the parameters of the spectral lines in each of the files, with the following format:

Column 1: Signal-to-noise ratio of the line

Column 2: Calibrated, integrated intensity of the line

Column 3: Full width at half maxiumum of the line. Used to calculate the uncertainty of the intensity

Column 5: Wavenumber of the transition in cm^{-1}.

Column 7: Lower level key of the transition. Must match the key in the lifetimes file.

Column 9: Upper level key of the transition. Must match the key in the lifetimes file.

An example is:

```
0.09 32000. 46000.
4      19090   130.04 3043.665   32855.131   II   d5____b4F4 - a3F4p_z4G5*
5      19630   117.26 3035.418   32944.389   II   3F4s__a4F2 - a3P4p_y4D1*
10     41560   142.82 3004.795   33280.144   II   3F4s__a4F3 - a3P4p_y4D2*
6      13610   110.76 3000.811   33324.329   II   d5____a2F3 - 3H4p__z2G4*
8      26890   121.93 2995.621   33382.062   II   3H4s__a4H4 - 3H4p__z4H3*
7      24410   162.60 2993.318   33407.744   II   3H4s__a4H5 - 3H4p__z4H4*
106   310900   150.52 2990.063   33444.107   II   3H4s__a4H3 - 3H4p__z4H3*
```

Xgremlin files: These are the .dat, .hdr, and .lin files used to make the linelist in Xgremlin. They are used for plotting individual lines within pybranch, without having to open up an Xgremlin window.

3 Interface

The main interface is shown in Figure 1. The left side of the interface has the buttons and entry fields for the files and levels of interest. The right side contains a scrolled text section with the results of the actions. The buttons and entry fields are as follows:

Quit button: Closes all files and quits the program.

Results button: First prints out the values of the residuals - i.e. the lines that were not observed in any spectra, but were included in the calculations. These are used to calculate a residual that is used in the calculation of the branching fraction. The program then prints out the Summary section, with the line wavenumber, lower energy level, weighting factor, measured branching fractions and uncertainty, transition probabilities and uncertainties, and the value in the theoretical calculations.

Log file:The content of the main window is written to this file. The extension 'log' is added to this name.

Upper Level: The upper level of interest. The format must be that used in the lifetimes file. Select the level from the drop-down list and then click the 'Upper level' button. This will find the level in the lifetimes and identified lines file and display it's lifetime, and all the previously-identified transition from it (First block in Fig. 1) . It will then get all the transitions in the branching ratio files from that level and display them in the main window (Second block in Fig. 1). It also populates the following buttons with the file names and lower levels.

Ref. Level: The lower level used for putting all of the intensities in the separate files on the same scale. The intensity of this line is set to 1000 in all the files and the other lines adjusted accordingly. The results are printed in the third block in the scrolled text field in Fig. 1.

Rescale File: This is the name of a file that does not contain the line to the reference level. The intensities in this file must be set using a Rescale level that gives a line in both the Rescale file and in at least one other file that contains the Reference level. A weighted average intensity and uncertainty is calculated from the line in all the other files that contain both the Rescale level and the reference level and the intensity of the line from the Rescale level is set to that value. The uncertainties of the intensities of all the lines in the Rescale file are then added in quadrature to the uncertainty of the weighted average intensity.

Rescale Level: The level used with the "Rescale File" entry.

Delete File: Used to delete lines in specific files that appear to be erroneous.

Delete Level: The level to delete in the "Delete File"

Plot Line to:This selects a lower level that can be plotted in the file "Select File"

in file menu:Calls up a file browser menu where the .dat file can be selected for plotting. If the .lin file is present in the same directory, the fit of the line is also plotted.

Add Comment:Adds a comment to the log file and the results window. Note that although it is possible to type a comment into the results window, those comments will not be saved in the log file, so the "Add Comment" button should be used instead to save any comments

4 Calculations and Uncertainties

The calculation of the branching fractions, transition probabilities, and their uncertainties follows the analysis of Sikström et al. [2002], with a few modifications from Ward et al. (2003). The transition probability between an upper level, k and lower level, i, is obtained by combining a measurement of the lifetime of the upper level τ_k with the branching fraction,

$$F_{ki} = \frac{I_{ki}}{\sum_j I_{kj}} \tag{1}$$

where I_{ki} is the intensity of the line in photons/s and the sum is over all of the decay channels from the upper level k. The transition probability A_{ki} is then calculated from τ_k using:

$$A_{ki} = F_{ki}/\tau_k \tag{2}$$

The following components of uncertainty are included in the PYBRANCH.PY script:

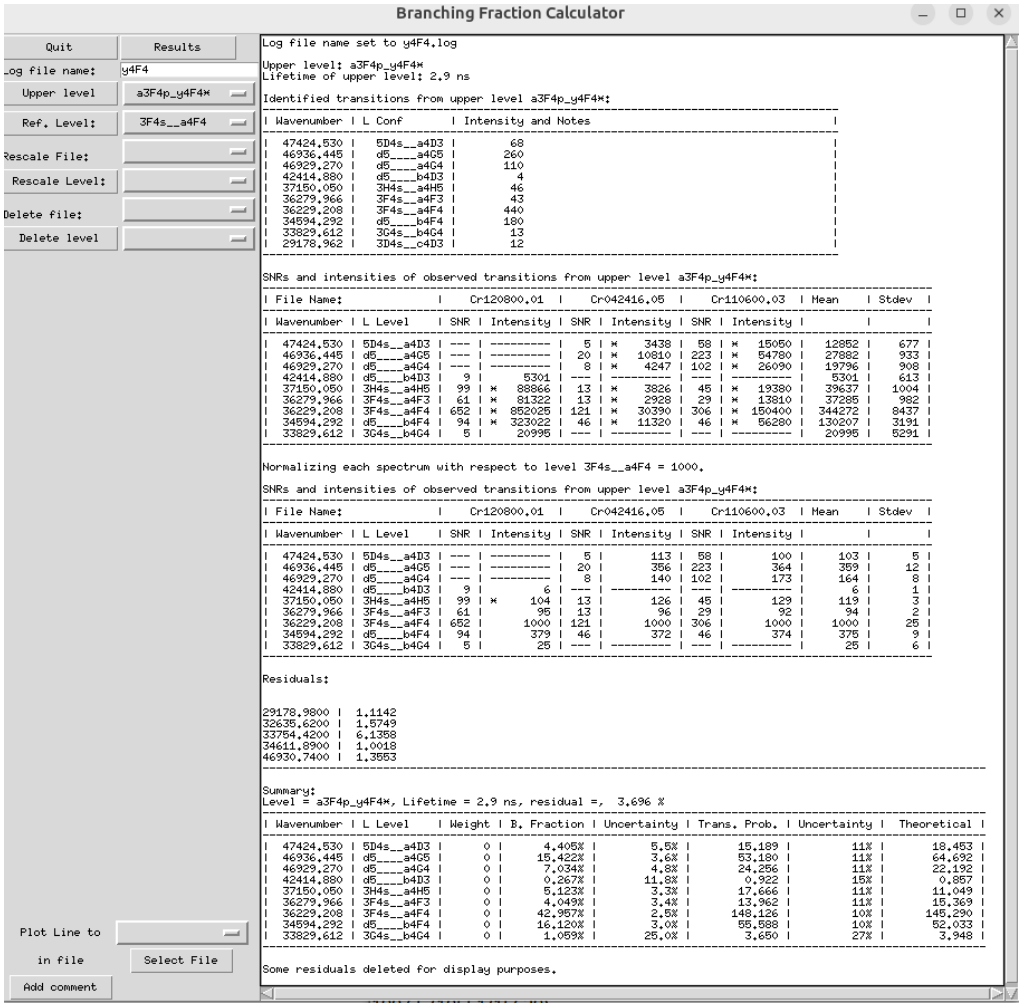
- The statistical uncertainty U_{stat} of the line intensity is estimated from equation 10 of Sikström et al. [2002] as:

$$U_{stat} = \frac{1.5}{(S/N)\sqrt{(n)}} \tag{3}$$

  where n is the number of points per FWHM. n is calculated from the resolution in the first line of the branching ratio files and the FWHM in the third column.

- The uncertainty with which the response of the spectrometer and imaging system can be measured is estimated from repeated observations of the calibration lamp on the same day. As noted in Section 3.2.1 of Sikström et al. [2002], applying this uncertainty directly to each line would be an overestimate, as lines close together in wavenumber will have a much smaller uncertainty in the calibration than lines well-separated in wavenumber. The program thus multiplies the uncertainty by a factor $\Delta\sigma/W_\sigma$, where $\Delta\sigma$ is the separation of the line from the strongest line from the upper level of interest, and W_σ is the total width of the observed calibration spectrum, given in the first line of each branching ratio file.

Figure 1: The pybranch.py interface



- The uncertainty of strong lines in each spectrum is limitied to a minimum of 6 % to ensure that strong lines receive similar weighting in the average. The 6 % minimum uncertainty was taken from the estimated calibration uncertainty and applied after combining the calibration and statistical uncertainties in quadrature.

- The inverse square of the uncertainty of the line in each spectrum is applied as a weighting factor to calculate F_{ki} for each spectral line. The uncertainty of F_{ki} is then then derived from the first two terms in equation 7 of Sikström et al. [2002].

- In cases where the reference line is not seen in one or more spectra, a transfer line is used to calculate a normalizing factor f to put the spectrum on the same intensity scale as the other spectra. The uncertainty of this normalizing factor $u(f)$ is derived from the weighted average of the quadrature sum of the uncertainty of the reference line $u(I_{ref,j})$ in each normalized spectrum, j, and the uncertainty of the transfer line $u(I_{trans,j})$ in each normalized spectrum:

$$u(f) = \sum_j \sqrt{u(I_{trans,j})^2 + u(I_{ref,j})^2} \tag{4}$$

  The uncertainty of the normalizing factor is added in quadrature to the uncertainty of each line in spectrum not containing the reference line.

- The residuals are applied as a correction to the lifetime before calculating F_{ki} and transition probability. The relative uncertainty of the residuals, is estimated as 50 %, as in Section 3.3 of Sikström et al. [2002]. As noted in Sikström et al. [2002], its influence on the uncertainties of the measured lines is small provided that the residuals themselves are small.

5 References

Sikström et al. (2002): Sikström, C. M., Nilsson, H., Litzén, U., Blom, A., & Lundberg, H. 2002, J. Quant. Spectrosc. Rad. Transfer, 74, 355

Ward et al. (2003): Ward, J.W., Li, J.J., Schwartz, J., Nave, G., Raassen, A.J.J., and Uylings, P.H.M. 2003, Astrophys. J. 959, 8