DARA Drift Scan Assignment

Job Vorster

14 April 2022

1 Coding Assignment

This assignment can be done with any single FITS file from the "Calibration" folder of HartRAO data.

1.1 Accessing the data

- Write a code to open the calibrator drift scan FITS file with `astropy.io.fits`.
- Write a code to access the following parameters from headers within the FITS file.
  - Date of Observation.
  - Astronomical source of interest.
  - Telescope name.
  - Nominal system temperature.
  - Centre frequency.

1.2 Noise diode calibration

Do all these steps for both polarizations.

- Get the noise diode temperature (with errorbar) from the header.
- Plot noise diode counts vs MJD.
- Calculate the mean background counts (the "off" counts) with `numpy.where` and `numpy.mean`.
- Calculate the average "on" counts with `numpy.where` and `numpy.mean`.
- Calculate the counts per K conversion factor.

1.3 Drift scan fitting

Do all these steps for the south, centre and north pointings. As well as for both polarizations.

- Divide the counts with the relevant counts per K conversion factor from the noise diode calibration.
- Use `numpy.where` to exclude the signal from the astrophysical source (i.e. the brightest Gaussian in the data).
- Fit a polynomial of 3rd order to the rest of the data points with `numpy.polyfit`.
- Subtract the polynomial (as a function of MJD) from the counts data to get "baseline corrected" data.
- Fit a Gaussian to the "baseline corrected" data with `scipy.optimize.curve_fit`.
- Store the values of the peaks, and their errors. *Hint: The pcov matrix from `scipy.optimize.curve_fit` contains the square of the errors on its diagonal.*
- After completing the baseline correction and Gaussian fitting for the north, centre and south pointings (and for both polarization). Write all of the peak values (in K) with errors in a Table, you will need it later.

1.4 Pointing correction

- Fit a Gaussian to the peak values of the north, centre and south scans as your y-values and [0,1,2] as your x values with `scipy.optimize.curve_fit`, you have to take the errors on the peaks into account. This is done by setting `sigma` equal to an array of your peak errors and `absolute_sigma=True` within `scipy.optimize.curve_fit`.
- Calculate the pointing correction by calculating the ratio between the Gaussian peak above, and the peak of the centre drift scan.
- Write down the pointing correction (with their errors) for both polarizations.

1.5 Flux density calculation

The last thing we need to do is to calculate the flux density of the source so that we can calculate the ratio between the real flux density and the antenna temperature to get the point source sensitivity (PSS).

- Get the paper: Ott M., Witzel A., et al., 1994, AA, 284, 331.
- Use the equation from Table 5 of the Ott et al., 1994 to calculate the "real" flux density of your astrophysical source.

- The ratio of the “real” flux density (in Jy) and the antenna temperature peak of the centre drift scan multiplied with the pointing correction is our final result. The PSS.

2 Report

This section describes a report you will write for this experiment. This section will be updated once we finish the coding tutorials.