Radio Astronomy Centre, Ooty

Pulsar Timing: Data Analysis

Pratik Manwani

Submitted to Dr. B.C Joshi

December 2017 -January 2018

Timing Analysis

1

Pulsar Timing is the regular monitoring of the rotation of the neutron star by tracking the arrival times of the radio pulses from the pulsar. Pulsar Timing can give information on many pulsar properties. For a pulsar which has been observed for a few years it is possible to calculate its period, period derivative, position, dispersion measure and many other parameters. Pulsar timing requires an initial timing model and a set of pulse arrival times for the process. These pulse arrival times are know as the 'Time of Arrivals'(TOAs). These TOAs were calculated by using data of a few days for the pulsar J1939+2134. They were then used to find its period and the period derivative.

1.1 Installation of the required software

The first part of analyzing the data was to download the required software packages. The packages used were:

1. PGPLOT
2. CFITSIO
3. FFTW
4. PSRCHIVE
5. TEMPO
6. TEMPO2

After installing, the paths have to be linked so that we can use the functions of the software directly, without specifying the path.

1.2 Analysis of the J1939+2134 Pulsar

The analysis was done using data for the pulsar J1749+2134. The data was already dedsipersed and the MJD was given at the start of each observation. The analysis was done as described by the following steps.The data provided was unfolded and contains RFI. The unfolded data was converted to time series form by using the command:

- reader <inpute-file> > <output-file>.asc

The plot of the time series data is shown in figure 1.1

The periods of folding was done using the the *SEEK* function of sigproc.

- seek <input-files>

The period to be used was the double of the period acquired from *SEEK* as the profile of the pulsar showed two pulses, the primary and secondary pulses. The secondary pulse originates about 180^o from the main pulse. The simplest interpretation is that it originates from the opposite magnetic pole of the neutron star.

Using these periods, the observation files were folded. Some of the plots of the folded profile are shown below.

A raw form of the period derivative could be found directly from the periods calculated from *SEEK* and the time(MJD) that was given in the files. The period derivative comes out to be $1.15940\ E-09\ s^-2$. This can be seen in figure 1.2.

The observations were than folded taking the period for each file to be the period calculated from *SEEK*. The folded profiles were different from each other as can be seen in the figure 1.3. Some of the pulses were not produced by *SEEK*. For

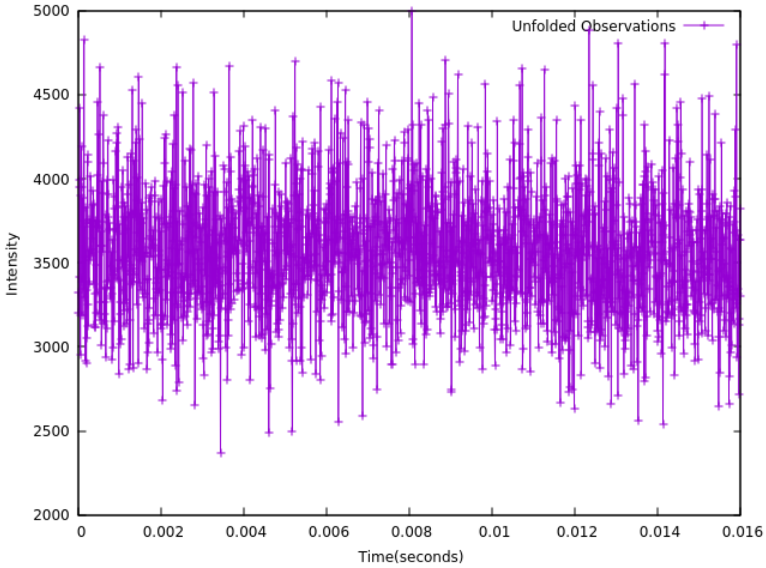


Fig. 1.1: The unfolded data

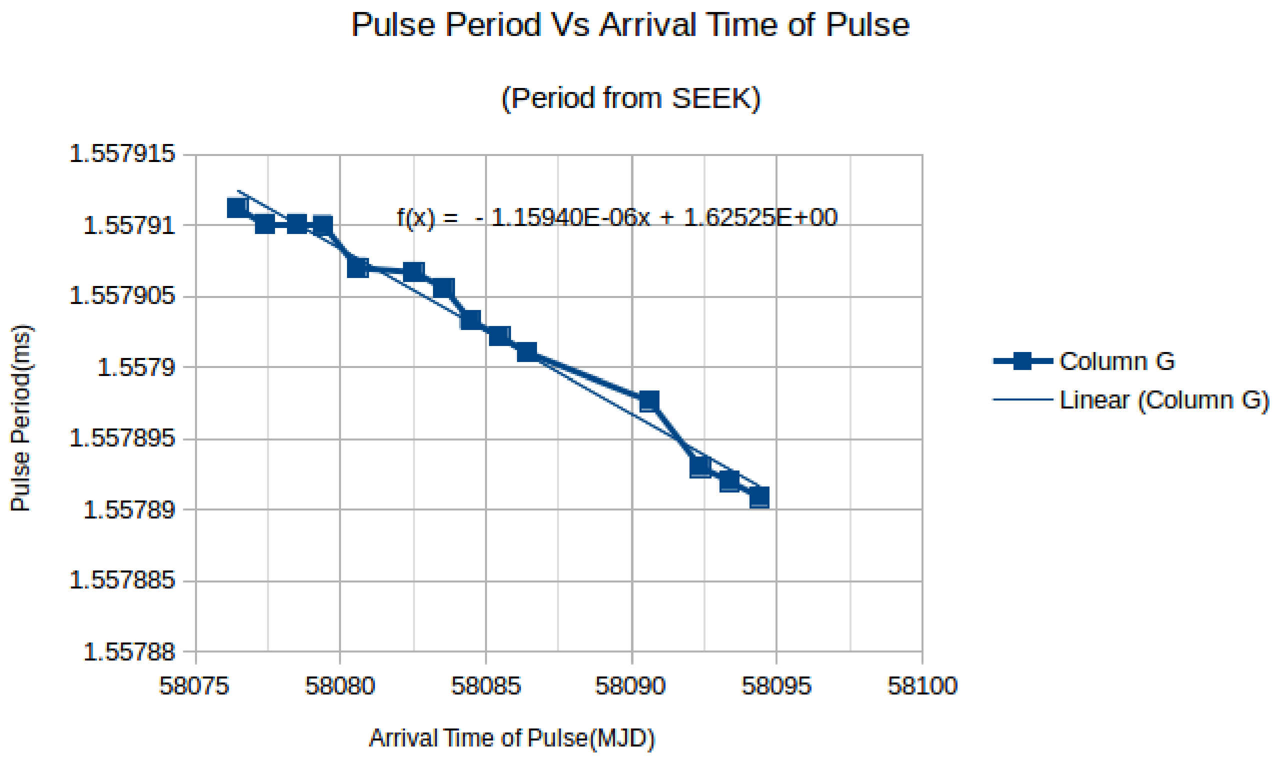


Fig. 1.2: Raw period derivative

these, the periods were calculated by folding near the assumed period to get the best period.

Some of the folded profiles are shown in figure 1.3.

Now the folded profiles are in .prof formats. We need to convert this into .fits formats.

After converting we can plot the fits file by using

- for file in *.fits
  do
  psrplot file -pD -jFp -D file.ps/cps
  done

To plot any of the files in .ps format, we use evince <file-name> And to put them in a pdf we can use:

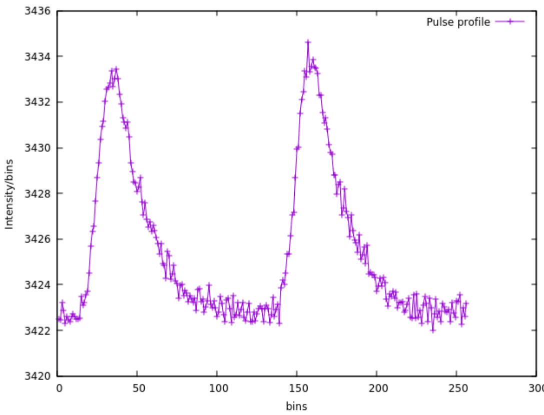
- gs -sDEVICE=pdfwrite -dNOPAUSE -dBATCH -dSAFER -sOutputFile = <name>.pdf *.eps

Using psrstat we can find the snr of all the files as shown below

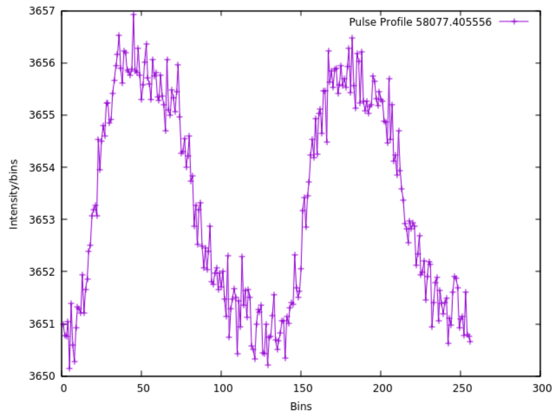
- psrstat *.fits | grep -e snr -e file

We get both the SNR and the filename. We need the profile having the highest SNR as the template. Now, we need to make the template noise free. To do this, we can fit multiple gaussians to the template. The following instructions describe the procedure to this.

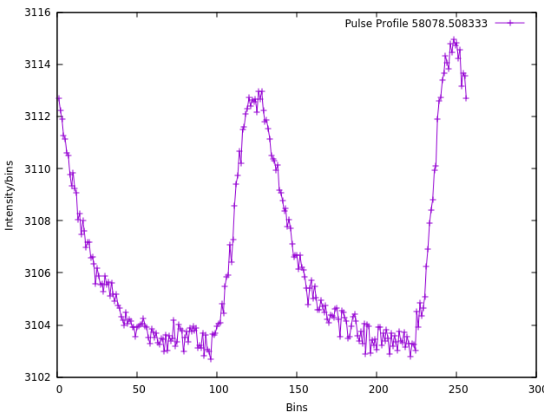
- paas -i <templatefile>.fits. A plot will open as shown in figure 1.4



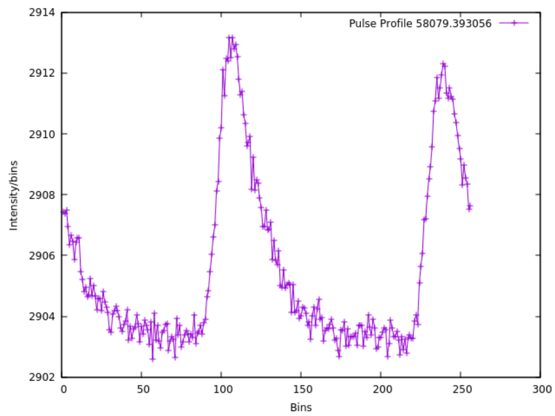
(a) MJD-58076.453472



(b) MJD-58077.405556



(c) MJD-58078.508333



(d) MJD-58079.393056

Fig. 1.3: The plots for the folded profiles using periods from SEEK

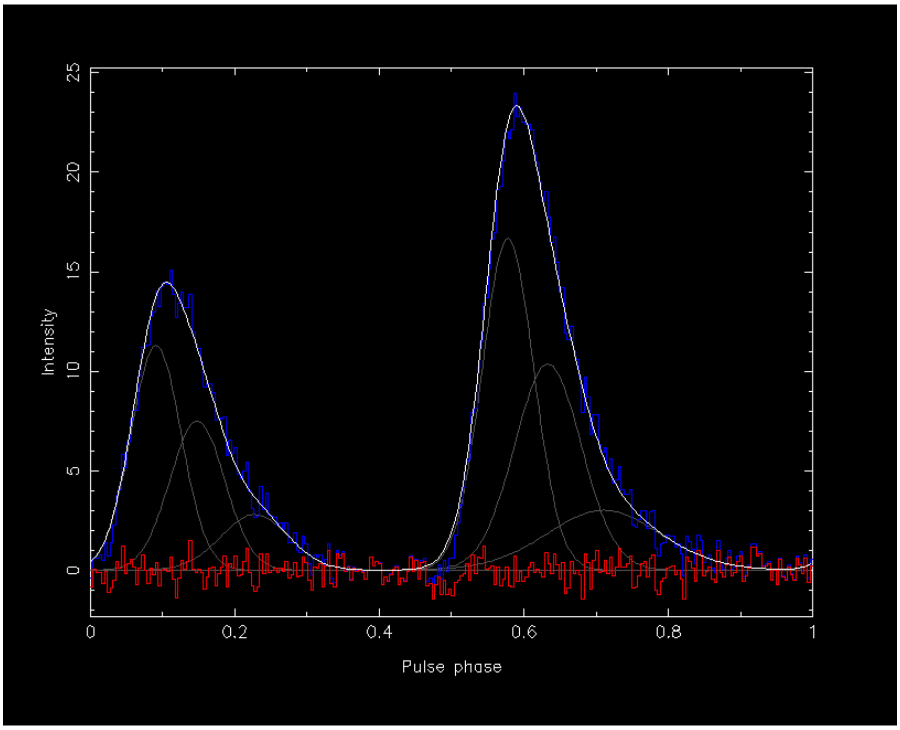


Fig. 1.4: Caption

- Click on plot to fix center of the Gaussian - Second click to choose half-width of the Gaussian - Third click at approximate height of the Gaussian model - Press 'f' on the plot key to fit

- You need to plot multiple gaussians till the unfitted(red) part goes below the baseline

- Press 'q' on the plot - The gaussian fitted template will be stored in a file name paas.std

The time of arrivals(TOAs) can be measured by cross correlating the data profile with this template which is representative of the shape of the pulsar. To get the TOAs we use the template and the PSRFITS file. The command for it is:

- pat -f tempo2 -A PGS -s paas.std(template) *.fits > <Toutput-file-TOAs>

Now, we need to find the timing residuals. The timing residuals are the differences between the measure pulses time of arrival and expected times of arrival.

- residual = observed TOA - computed TOA

For the computed TOA we need the parameter file of the pulsar. We can get this file from either the ATNF catalogue or the IPTA data release. The parameter file will be in .par format. Now we can use it and the observed TOAs 1.2 to get the timing residuals. To get them we need to use Tempo2.

- tempo2 -gr plk -f <par-file <TOA-file> -nofit

This produces a plot of the pre-fit residuals. Pressing '2' on the plot gives the post-fit residuals. The -nofit command switches off the fitting algorithms. To fit for a given set of parameters, we can select those from the above panel in the window. Some of the files containing high variation in timing residuals were removed by right clicking the points on the plot. For the analysis, since the data was for a few days, the residuals were fit for $F0$ and $F1$. The timing residuals are shown in the figure 1.5. The value of F0 and F1 from the fit are 641.92816260 and 1.89152115e-13 respectively. The true values from the parameter file are 641.92822458 and -4.33127347e-14 respectively. The rms value of the residuals is 115.584 μs.

After fitting the residuals, the new parameter file containing the post-fit parameters can be generated.

Tempo2 produces "polyco" files that contain the pulsar parameters in the form of a simple polynomial expansion. To generate it we use the command:

- tempo2 -f <new-par-file> -polyco "<MJD-start> <MJD-end> nspan ncoff maxha ort<site-code> 326.5<site-freq>" -tempo1

The code above will request tempo2 to make a polyco file for the pulsar from start MJD to end MJD with the file being divided into segments each of $nspan$ minutes.

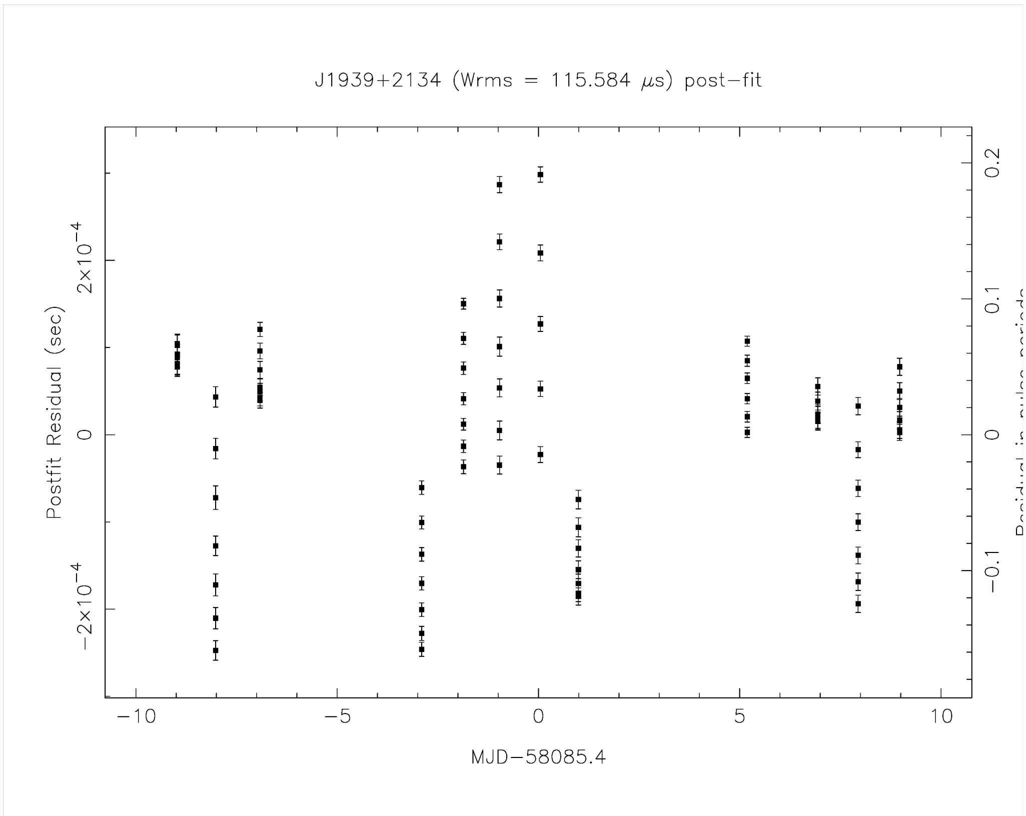


Fig. 1.5: The plot for the post-fit timing residuals

The number of coefficients for use is determined by ncoff. The maximum absolute hour angle for the prediction, generally set at maxha = 8. The observatory site code and observing frequency also have to be set.

Using this polyco file, we can fold the data. This improves the SNR of the profile drastically because now the periods are set according to the polyco file which has the fitted parameters for both, the period and its derivative.

To fold the data with the polyco file, we use the following command:

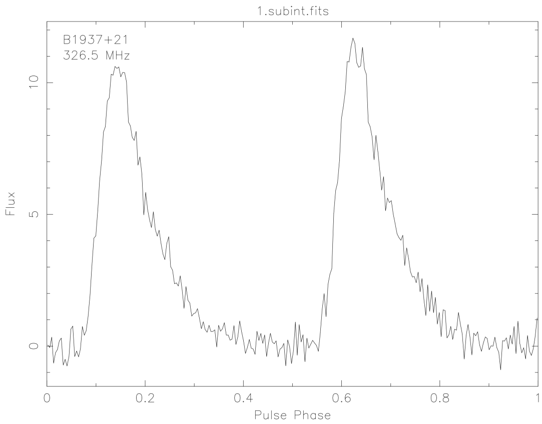
- fold <input-files.tim> -p <polyco-file.dat> -n 256 > <output-file>

The folded profiles will be in .prof format. We need to convert it to .fits format.

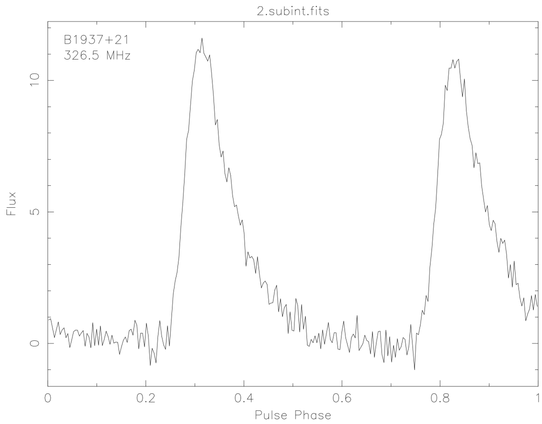
We can plot these files using the for loop in 1.2 to check the profiles that we have generated. The plots for some of the profiles can be seen in the figure 1.6. The SNR values have improved and the broad peaks have narrowed down.

We have to follow the steps again to generate the timing residuals from this new fit. The steps include template making, calculating TOAs and using the pulsar parameter file to get the timing residuals.

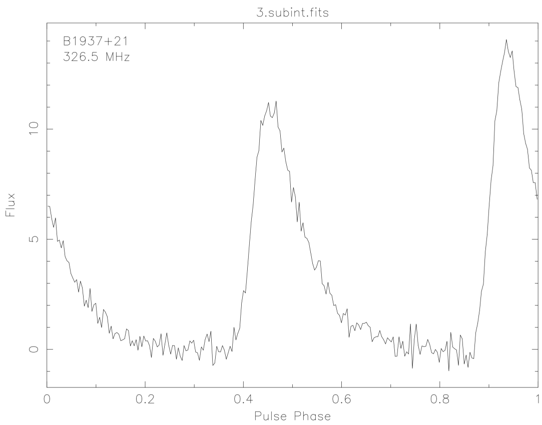
The values of F0 and F1 from this fit are 641.92822424 and -4.20332471e-14. These are closer to the true values from the parameter file of 641.92822458 and -4.33127347e-14. The rms of the timing residuals reduced by a factor of 100 to 1.216 μs as shown in figure 1.7.



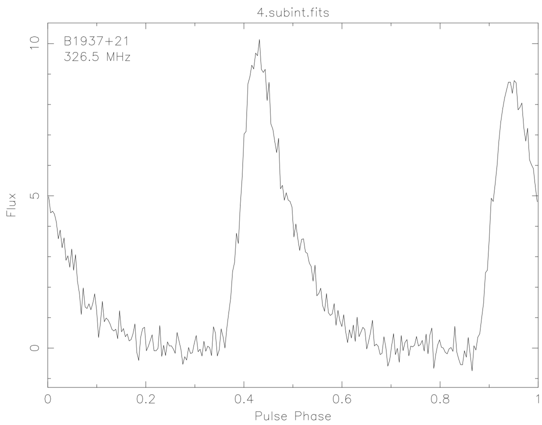
(a) MJD-58076.453472



(b) MJD-58077.405556



(c) MJD-58078.508333



(d) MJD-58079.393056

Fig. 1.6: The plots for the folded profiles after using polyco

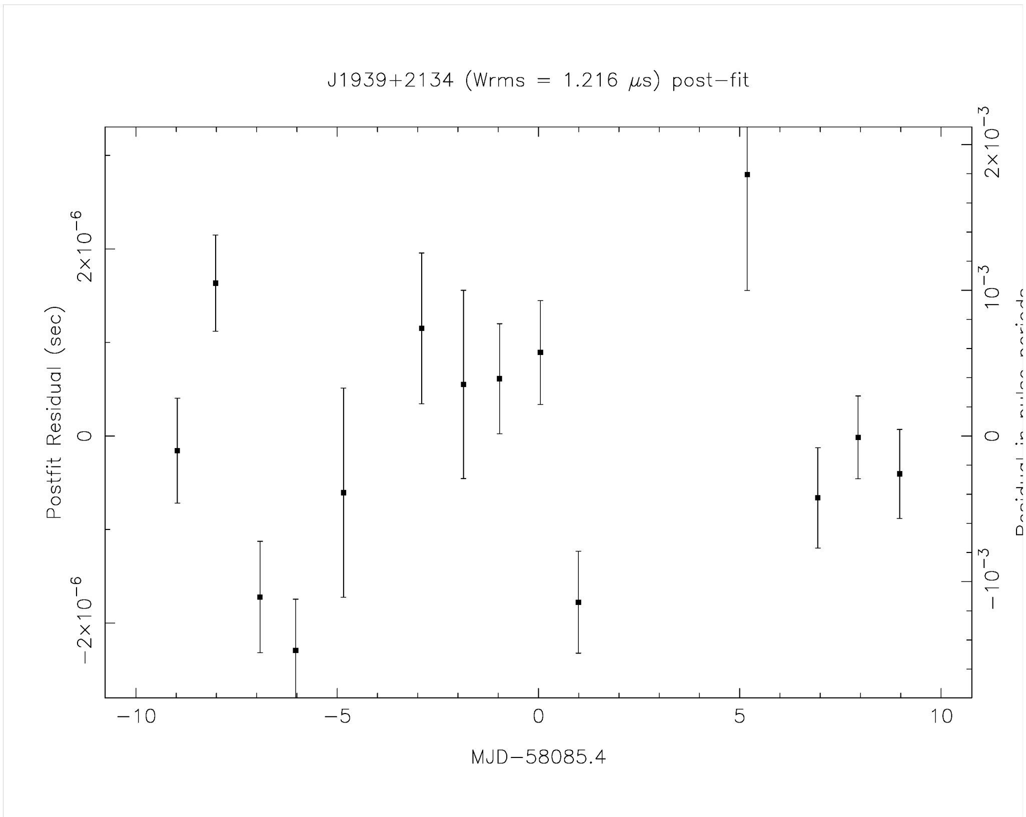


Fig. 1.7: The plot for the post-fit timing residuals after using polyco to fold the profiles

Dedispersion | 2

The electrons in the interstellar medium disperse the pulsed signal received by a radio telescope such as Ooty Radio Telescope (ORT) from a radio pulsar. There is a broadening of an otherwise sharp pulse when a pulsar is observed over a finite band width. The amount of dispersion is proportional to the integrated column density of free electrons between an observer and the pulsar. A measure of this integrated column density of free electrons is called the Dispersion Measure (DM). Assuming the density of electrons in the inter-stellar medium (ISM) to be uniform, this quantity depends on the distance of the pulsar. Thus, if the DM can be measured, so can be the distance to the pulsar.

$$DM = \int_0^d n_e dl \tag{2.1}$$

where, n_e is the electron density

The pulse over one channel (with reasonably small bandwidth) can be assumed to be similar in shape to the pulse emitted by the pulsar and the time delay of this pulse between the highest and the lowest frequency channel can be used to estimate the dispersion delay (DM or distance).

The simplest way to compensate for the effects of pulse dispersion is to split the incoming frequency band into a large number of independent frequency channels.

The appropriate time delay has to be calculated using the following equation:

$$\Delta t \simeq 4.15 \times 10^6 ms \times (f_1^{-2} - f_2^{-2}) \times DM \tag{2.2}$$

where, Δt is the difference in the arrival time of the pulse at the frequencies, f_1 and f_2.

2.1 Analysis of the Pulsar B1749-28

The analysis was done using data for the pulsar B1749-28. The analysis was done as described by the following steps.

The observations were in the filterbank format .fil. So we can use the sigproc function $dedisperse$ to dedisperse the data into corresponding bands.

- dedisperse <input-file>.fil -d 0.0 -b <no. of bands> > <output-file>.sub.tim

where, -d is the dispersion measure which is to be used. In our case we do not have the DM so we set it to 0. The number -b is taken to be 8 which divides the 16 MHz band(318.5 to 334.5 MHz) into 8 sub bands of 2 MHz each.

For the analysis, the period was taken by using the SEEK function of sigproc. The function was used on the data from the full band without dedispering. This can be done since SEEK analyzes in the frequency domain so the spin frequency would just be phase shifted for all the observing frequencies.

- fold <input-file>.tim p <period> n <bin number> nobaseline > <output-file>.sub.prof

For this pulsar, the period from SEEK comes out to be 562.536645076 ms. The bins taken are 256.

Since the pulses observed at higher frequencies arrive earlier at the telescope than their lower frequency counterparts, there is a time delay between the pulses. The pulses observed at different bands are shown below.

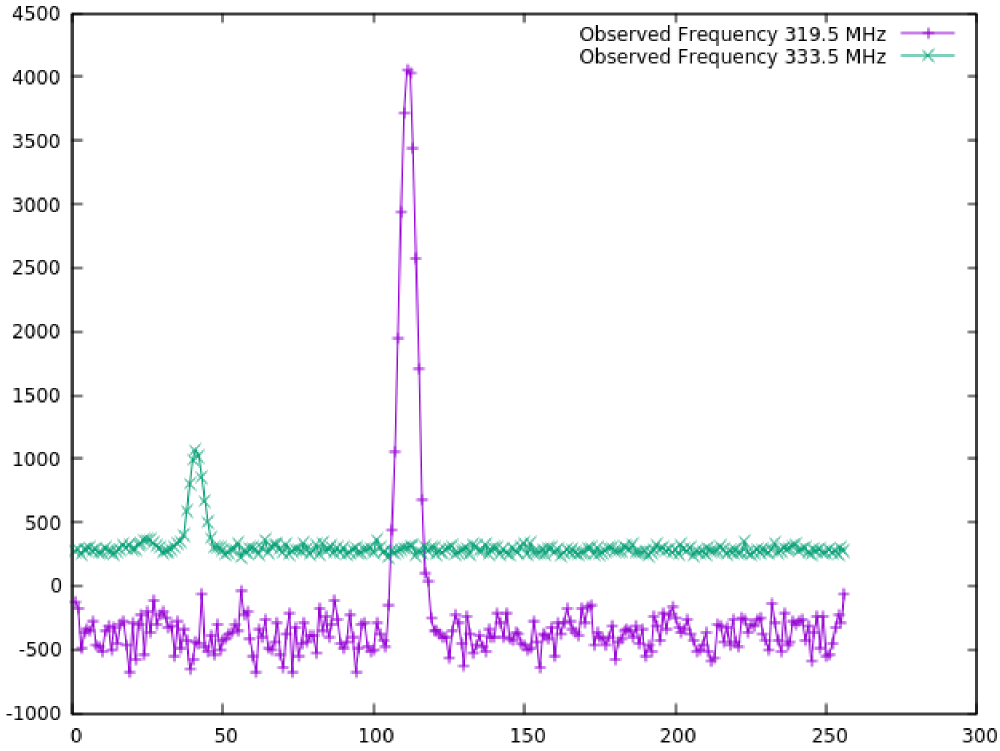


Fig. 2.1: The pink coloured plot corresponds to the sub band centered at 319.5, while the green coloured plot corresponds to the sub band centered at 333.5. As you can see there is a time delay between the sub bands.

To get the exact position of the pulses we need to fit the peaks with Gaussians using GNUPLOT. To do this we need to define the gaussian function as:

- f(x) = a*exp(- ((x-b)**2/(2*c**2)))

We need to give the average values of a,b,c to fit it to the pulse profile

- fit f(x) '<output-file>.sub.prof' via a,b,c

The values of b(centre of the peak) are noted down. The values obtained for all the pulses are written down in table below.

We can get the time of arrival of the pulses from the bin numbers and plot the corresponding graph between the observing sub bands and time of arrival.

Frequency	TOA of Pulse(bin)	TOA of Pulse(ms)	Error(bin)	Error(ms)
319.5	111.326	244.6287287099	0.05264	0.1156715976
321.5	101.926	223.9730862735	0.0803	0.1764519242
323.5	90.8999	199.7442374365	0.2052	0.4509082796
325.5	80.2467	176.3348023298	0.05614	0.1233625283
327.5	69.0891	151.8169942395	0.09619	0.2113687496
329.5	58.5923	128.7512338644	0.09092	0.1997884054
331.5	49.3331	108.4049865828	0.07818	0.1717934176
333.5	41.1841	90.4983025175	0.04853	0.1066402476

Tab. 2.1: The table listing the time of arrival of the pulses for the corresponding observed frequency

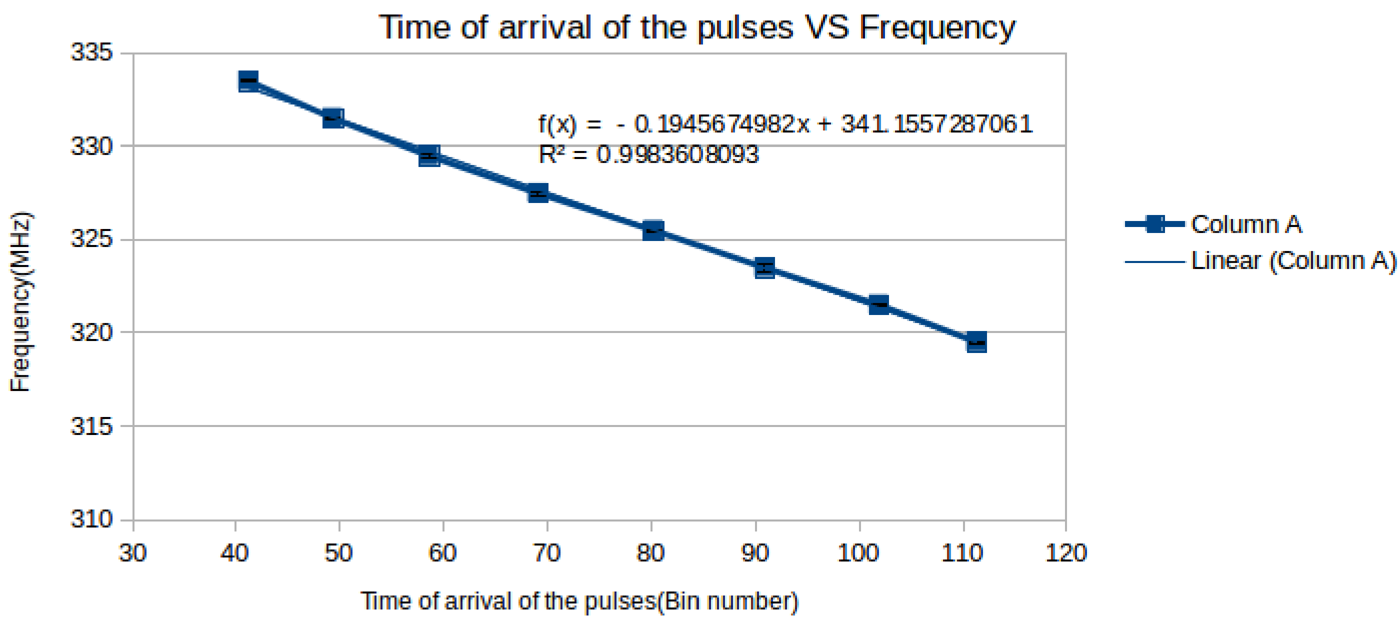


Fig. 2.2

Now we can use equation 2.2 to get the value of the dispersion measure. Using the values obtained for $f_1 = 319.5$ and $f_2 = 333.5$, we get $\delta t = 154.1304261924$. Putting these values, in the equation we get the dispersion measure,
$DM = 46.1246 \pm 0.0665 pc/cm^3$.

To get a better result we can use the slope of the trendline and use it to determine the dispersion measure.

We can use this form of the equation 2.2.

$$t_1 - t_2 = 4.15 \times 10^6 \times \frac{(f_2 + f_1) * (f_2 - f_1)}{f_1^2 * f_2^2} \times DM \tag{2.3}$$

And convert it to:

$$\frac{1}{DM} = 4.15 \times 10^6 \times \frac{(f_2 + f_1)}{f_1^2 * f_2^2} \times -Slope \tag{2.4}$$

Using equation 2.4, we get the dispersion measure as
$DM = 47.3166 \pm 0.0061 pc/cm^3$.

The values that we got our close to the actual dispersion measure i.e $50.88 pc/cm^3$