# IPTA 2015 Student Workshop: Polarimetric Calibration Using PSRCHIVE

Willem van Straten & Matthew Kerr

July 22, 2015

## 1 General introduction

The PSRCHIVE programs to be used as part of this tutorial print a brief help message when the `-h` command line option is used; e.g. `psredit -h`. Online reference manuals are also available at

`http://psrchive.sourceforge.net/manuals`

The PSRCHIVE software deals with observational data stored as a three-dimensional array of pulse profiles; the axes are time (sub-integration), frequency (channel), and polarization. The physical properties of the data are described by various attributes called meta-data. This tutorial introduces some basic features of pulsar data analysis using PSRCHIVE, ending with a focus on tools specific to polarimetric calibration and pulsar timing.

### 1.1 Set up your virtual machine

To begin the tutorial, log in to the pulsar virtual machine, `pulsarVMlite v1.05`; e.g.

```
ssh -X -p 2222 pulsar@localhost
```

Assuming that your shared folder is in `$HOME/share`, run the following commands to make a copy of the tutorial data and create a useful environment variable

```
cp -R $HOME/DATA/psrchive_data/mem $HOME/share
export PSRCHIVE_DATA=$HOME/share/mem
```

If you have not had much luck with adding a shared folder to your virtual machine, then you can make more room in the `$HOME/DATA` directory by deleting a few sub-folders and working in this directory instead; e.g.

```
rm -r $HOME/share/mem
rm -r $HOME/DATA/psrchive_data/zap
rm -r $HOME/DATA/psrchive_data/pca
export PSRCHIVE_DATA=$HOME/DATA/psrchive_data/mem
```

## 2 Query and modify meta-data with psredit

In the $PSRCHIVE_DATA directory, the receiver name is not set in any of the data files (*.ar). Set the receiver name to MULT_1 using psredit and the *output option* to overwrite the original files.

```
cd $PSRCHIVE_DATA
psredit -c rcvr:name=MULT_1 -m *.ar
```

The data files in $PSRCHIVE_DATA are a mixture of three different types. Create two sub-directories called pulsar/ and cal/ then use psredit to query the type attribute and use this to sort the files into the two sub-directories. (All three calibrator file types can go into the cal/ sub-directory.)

```
cd $PSRCHIVE_DATA
mkdir pulsar/
mkdir cal/
psredit -c type *.ar | grep Pulsar | awk '{print $1}' > psr.ls
awk '{print "mv "$1" pulsar/"}' psr.ls | sh
mv *.ar cal/
```

## 3 Plotting data with psrplot

Use psrplot to plot the Stokes parameters integrated over all frequency channels for each file.

```
psrplot -p stokes -jF pulsar/*.ar
```

The white line is the total intensity; red, green, and blue correspond to Stokes $Q$, $U$, and $V$, respectively. The Stokes parameters vary with time owing to the rotation of the receiver feed with respect to the sky (the parallactic angle). This effect is exploited to model the polarization cross-coupling in the instrumental response and calibrate the data.

## 4 Calibrating data with pac

A first-order approximation to calibration can be performed using observations of a noise diode that is coupled to the receptors, from which the complex gains of the instrumental response as a function of frequency are derived. This approximation to calibration is based on the *ideal feed assumption* that the Jones matrix in each frequency channel has the form

$$\mathbf{J} = \begin{pmatrix} z_0 & 0 \\ 0 & z_1 \end{pmatrix} \tag{1}$$

where $z_0$ and $z_1$ are the complex gains. That is, it is assumed that there is no cross-coupling between the receptors.

## 4.1 Excision of frequency channels that are known to be corrupted

Although the observations used in this tutorial are not adversely affected by radio frequency interference, the data near the edges of the band are known to be corrupted by quantization distortions. This can be seen using the loop-over-index option to plot the total intensity profile (the plot type named `flux` or its short-cut `D`) as a function of frequency in one pulsar data file; e.g.

```
psrplot -pD -l chan=0- -jp pulsar/n2003200180804.ar
```

Note that `chan=0-` specifies the entire range without having to know the index of the last frequency channel. The pulse profile is significantly distorted at the edges of the band due to a quantization effect called "scattered power" that arises during analog-to-digital conversion using 2 bits/sample.

To mitigate the impact of scattered power, use the `psrsh` command language interpreter and the `zap edge` command to assign zero weight to 15% of the total bandwidth ($\sim$ 19 frequency channels) on each edge of the band. Use the *output option* to write output data files with a new extension.

```
cd $PSRCHIVE_DATA
psrsh - -e zz pulsar/*.ar cal/*.ar << EOD
zap edge 0.15
EOD
```

In the above example, the single hyphen (`-`) command-line option instructs `psrsh` to read the command script from the standard input.

## 4.2 Display calibrator parameters with `psrplot`

In PSRCHIVE, calibrator observations of the noise diode have `type=PolnCal` (as returned by `psredit`); the noise diode is typically driven by a square wave with a 50% duty cycle, as can be seen by plotting the pulse profile of each calibrator observation

```
cd $PSRCHIVE_DATA/cal
psrplot -p D -jFTp *.zz
```

Using the command line option to loop over an index, plot the phase-vs-frequency images of $\Re[AB^*]$ and $\Im[AB^*]$ (`pol=2,3`), which effectively correspond to Stokes $U$ and $V$.

```
cd $PSRCHIVE_DATA/cal
psrplot -p freq -l pol=2,3 *.zz
```

These quantities vary with frequency due to an instrumental effect; the two orthogonal polarizations propagate through different signal paths with slightly different lengths, introducing a relative phase delay that varies linearly with radio frequency.

The absolute phase of the Jones matrix is lost during detection; therefore, the ideal feed approximation may be parameterized using a polar decomposition described by the absolute gain $G$, differential gain $\gamma$, and differential phase $\phi$. Plot the polar decomposition of the ideal feed as a function of frequency.

```
cd $PSRCHIVE_DATA/cal
psrplot -p calm *.zz
```

## 4.3 Preparing flux calibrator data with `fluxcal`

Although not immediately necessary for pulsar timing, it is also useful to perform absolute flux calibration, which may be later useful in other studies that could have an impact on long-term timing analyses; e.g. studies of refractive scintillation.

Absolute flux calibration information is derived from observations made while pointing at or near a standard candle such as Hydra A; these observations have a `type` attribute equal to `FluxCalOn` or `FluxCalOff`. Start by creating a calibrator database. In the `cal/` sub-directory, run

```
pac -w -u zz
```

to create a file called `database.txt`. Then run

```
fluxcal -f -d database.txt
```

This will produce a file named `n2003201035947.fluxcal` and update `database.txt` with a new entry for this file. Use `psrplot -p calm` to plot the derived estimates of the system equivalent flux density $S_{sys}$ and calibrator flux $S_0$ as a function of radio frequency.

## 4.4 First-order calibration

To perform the first-order approximation to calibration based on the ideal feed assumption, run

```
cd $PSRCHIVE_DATA/pulsar
pac -d ../cal/database.txt *.zz
```

For each input file, a new output file will be created with a new extension, `.calib`. If the receiver were ideal, the first-order approximation to calibration would have eliminated the variation of the Stokes parameters as a function of parallactic angle. Use `psrplot` to test this expectation.

```
psrplot -ps -jF *.calib
```

Why do the Stokes parameters still vary as a function of time?

Create a sub-directory, e.g. `ideal/` and move the newly calibrated data to this sub-directory (otherwise, they will be over-written in the next step).

```
mkdir ideal
mv *.calib ideal/
```

## 4.5 Measuring the cross-coupling parameters with `pcm`

To properly calibrate these data, the cross-coupling terms (off-diagonal components of the Jones matrix) must be estimated, which can be done by modeling the Stokes parameters as a function of time. The process of performing the least-squares fit is called Measurement Equation Modeling (MEM) and the PSRCHIVE implementation is described in van Straten (2004, ApJS 152:129).

First, use `psradd` to combine the first-order calibrated archives into a single archive

```
psradd -T -o calib.TT ideal/*.calib
```

The resulting file `calib.TT` is the first guess at the calibrated pulse profile; it is provide to `pcm` so that is can choose the best pulse phase bins to use as constraints.

Next, run `pcm` to derive the cross-coupling parameters, which are output in a file named `pcm.fits`.

```
pcm -d ../cal/database.txt -s -c calib.TT *.zz
```

What is the `-s` option and why is it used?

It will take about ten minutes to perform the least-squares fit in each frequency channel. On a multi-processor machine, multiple channels may be solved simultaneously by using the `-t <nthread>` command-line option, where `<nthread>` is the number of processing threads to run in parallel. If you are running out of time, you can skip this step and use the previously computed solution in `$PSRCHIVE_DATA/pcm.fits`.

While running, `pcm` outputs messages about the quality of the least-squares fits, which are performed independently in each frequency channel. When `pcm` finishes, it produces an output file `pcm.fits` that contains the MEM solution; the model parameters may be plotted using

```
psrplot -p calm pcm.fits
```

Compared to the solution derived using the ideal feed assumption, three new parameters have been added to the model of the receiver: $\theta_1$ describes the non-orthogonality of the feed receptors (the linearly polarized receptors should be oriented at 0 and 90 degrees) and $\epsilon_k$ are the ellipticities of

the receptors, which should be 0 in an ideal feed with linearly polarized receptors. The mean value of $\sim 5$ degrees corresponds to roughly 15% mixing between linear and circular polarizations (Stokes $Q$ and $V$). It is not possible to determine the absolute rotation of the receptors about the line of sight, $\theta_0$, without an external reference; therefore, only the non-orthogonality is measured.

The solution output by `pcm` also includes estimates of the Stokes parameters of the noise diode, which is no longer assumed to illuminate both receptors equally and in phase. This information enables the calibrator solution derived from one data set to be applied to observations of another source.

### 4.6 Full calibration

Move the file `pcm.fits` to the `cal/` sub-directory, change to this directory, and recreate the calibrator database

```
mv pcm.fits ../cal/
cd ../cal
pac -w -u zz -u fits -u fluxcal
```

Confirm that `pcm.fits` has been added to `database.txt`, then run

```
cd ../pulsar
pac -d ../cal/database.txt -S *.zz
```

Plot the Stokes polarization profile (integrated over the entire band) in each calibrated data file output by `pac` to confirm that they no longer vary with time.

## 5 Producing arrival time estimates with `pat`

In this section, arrival time estimates are derived using a previously created standard (template) profile. This high $S/N$ standard profile was formed by adding data from 5 different days (about 42 hours of integration).

### 5.1 Preparation of the standard profile

The standard profile is located in `$PSRCHIVE_DATA/std/standard.ar`. Plot the phase-vs-frequency image of the total intensity and compare this image with that of `nonspc.ar` in the same directory. The file `nonspc.ar` was formed from data that were not corrected for scattered power. Although `standard.ar` was corrected, there still remain residual artifacts in the edges of the band. Use `paz` to give zero weight to the affected frequency channels. For best results, excise the same frequency channels that were excised from the data in Section 4.1.

Use `pam` to integrate over all frequency channels; e.g.

```
pam -F -e FF standard.zz
```

then use the wavelet smoothing algorithm implemented by `psrsmooth` to create a "noise-free" template profile

```
psrsmooth -W -t UD8 standard.FF
```

This will produce a file called `standard.FF.sm`. Use the `crop` attribute of the `flux` plot to zoom in on the low amplitude flux near the off-pulse baseline and compare the standard profile with its smoothed version; e.g.

```
psrplot -pD -jp -c crop=0.01 -N 1x2 standard.FF standard.FF.sm
```

## 5.2 Estimation of arrival times

In this tutorial, we will compare two different methods of arrival time estimation, both of which work in the frequency domain: scalar template matching using only the total intensity, and matrix template matching using the full-polarization profile.

We also have two standards with which to experiment: the smoothed and not smoothed versions of `standard.ar`.

Finally, there are three different data sets: the uncalibrated data, the data calibrated using the first-order approximation, and the data calibrated using the solution derived with `pcm`.

All together, there are 12 combinations of arrival time estimation algorithm, standard profile, and data. Experiment with these combinations to find the arrival times with the lowest residual standard deviation.

To experiment, you will need to run `pat` in either scalar template matching mode; e.g.

```
pat -F -s standard.FF *.zz > uncalibrated_unsmoothed_stm.tim
```

or matrix template matching mode

```
pat -F -p -c -s standard.FF.sm *.calib > calibrated_smoothed_mtm.tim
```

You will also need to run tempo2 to evaluate the arrival times; e.g.

```
tempo2 -f pulsar.par calibrated_unsmoothed_stm.tim
```

Search the output of `tempo2` for lines like

```
RMS pre-fit residual = 0.108 (us), RMS post-fit residual = 0.108 (us)
Fit Chisq = 359.7       Chisqr/nfree = 359.74/95 = 3.78671      pre/post = 1
```

and make note of both the **RMS post-fit residual** and **Chisqr/nfree** in each case tested.

## 5.3 Discussion

Form groups to compare the arrival time analyses and consider questions such as

- Why is the reduced $\chi^2$ so far from unity in each case?
- What is the effect of using the smoothed standard profile?
- Does the smoothed standard profile have the same effect on both scalar and matrix template matching?
- What is the **last harmonic** reported by the matrix template matching algorithm? (Hint: plot the *Stokes; fluctuation power spectra* of the standard profile using `psrplot`)
- Is it necessary to calibrate the data before applying matrix template matching?
- Does calibration always improve the arrival time precision?