VLBA Sensitivity Upgrade Memo 32

DiFX Pulsar Guide

Walter Brisken & Adam Deller

December 5, 2010

Abstract This document briefly explains the pulsar capabilities of the DiFX software correlator and provides instructions, from both user and operator perspectives, on its use with a focus on the incarnation of DiFX as the VLBA correlator. The pulsar capabilities of DiFX include binary gating, binning, and matched-filter gating. This guide does not attempt to teach fundamentals of pulsar timing or document standard pulsar software. This guide is written for DiFX version 2.0.0, however, most of the document applies without change to 1.5.x series versions as well.

1 Pulsar capabilities within DiFX

The periodic nature of pulsar emission, usually with very deep, relatively long periods of no emission, offers the opportunity for increase in mean sensitivity of pulsar observations by weighting accumulations on a very short timescale (given by a single FFT of the FX correlator, or $\sim 1\mu$s). Alternately, the option exists to generate phase-resolved integrations, generating multiple output visibility databases, each representing a distinct pulse longitude (or phase) range. These two capabilities are called gating and binning, respectively and are described below.

A final pulsar capability of DiFX is called *profile mode*. In this mode, the correlator is configured to only generate binned auto-correlations, in a manner analogous to the online folding performed by typical pulsar backend instruments, with the intention of using these phase bins to derive the pulsar spin phase.

For the DiFX correlator to weight visibilities correctly in pulsar mode, it requires both the ability to predict the pulse longitude at any given frequency and time in the observation, and the knowledge of the desired weight to be applied to any given pulse longitude. The pulse phase prediction information is supplied by way of a predictive polynomial file (a "polyco" file, which is generated using pulsar timing software such as `tempo` or `tempo2`, described in Section 2.1) and the weighting information is supplied in the DiFX "binconfig" format described in Section 2.2.

Note that pulsar processing is incompatible with some other DiFX capabilities such as simultaneous correlation at multiple phase centers.

1.1 Gating

Generalized pulsar gating allows a pulse-phase-dependent weight to be applied during the accumulation step of the cross correlation computation. DiFX implements this generalized mode with the more common *binary* gating being a special case. During gating observations, the option `SCRUNCH OUTPUT` in the pulsar bin configuration file (see below) should be set to true. Pulse longitude is divided into segments by providing the end phase (a number between 0 and 1 corresponding to the fraction of rotation from the ephemeris reference time); there is no requirement that the longitude segments are uniformly spaced, but they must appear in order of increasing pulse phase. Each segment is assigned a weight (a number between 0 and 1, inclusive). In binary gating, two segments are defined with one segment set to a weight of 0 and one segment set to a weight of 1. Note that it is traditional to use pulsar timing software to produce predictive `.polyco` files which place the peak of the pulse at around zero phase, but such an outcome depends on the pulsar ephemeris used to generate the `.polyco` files, and it is the responsibility of the PI to ensure that the provided gate parameters match the pulse profile produced by the provided `.polyco` file. It is required, however, that the segments are chosen in a manner consistent with the zero phase of the pulse ephemeris; it is the experiment PI's responsibility to arrange for this.

Matched-filter pulsar gating offers the greatest gain in sensitivity. In this mode of operation, each segment should be assigned a weight that is proportional to the pulsar brightness at that pulse longitude. In most cases the gain in matched-filter gating over binary is very small, up to about 50% in good cases, and presents greater opportunity for degrading performance in cases where the pulsar ephemeris is not known accurately. Usually most of the gain in matched-filter gating can be had with about 15 pulse bins, except perhaps in cases where the pulse profile consists of many maxima.

In any pulsar gating mode, the FITS-IDI file that is ultimately produced will be very similar to that produced for non-pulsar observations.

1.2 Binning

Pulsar binning is arranged in the same manner as (generalized) pulsar gating with two exceptions: `SCRUNCH OUTPUT` is set to false (preventing the weighted averaging across the pulse profile) and the provided weights are used to scale the amplitudes; it is thus recommended to set the weight to 1.0 for all bins in this mode unless there is a particular reason that unnatural scaling is desired.

One use case of binning is for correlation of pulsars whose spin ephemeris is completely known with the exception of the absolute phase. In this case uniform segment widths are usually used. The number of segments needed depends on the fractional pulse width.

Unlike for gating, data from each separate pulsar bin gets written to a separate output FITS-IDI file; non-pulsars correlated within the same correlator jobs will only be stored in the first bin (bin 0) output file. The potentially large multiplier in output storage (and intermediate correlator communications) presents some negative consequences, especially for large (> 20) numbers of bins. Currently AIPS has no capability to *scrunch* multiple phase bins (i.e., perform generalized gating post-correlation). Please consider these downsides when planning pulsar binning observations and do not use pulsar binning as a replacement for pulsar gating when that is possible.

1.3 Profile mode

Normally autocorrelations from DiFX are not generated separately for each phase bin. A special `profile` mode has been implemented that configures the correlator to produce pseudo-cross-correlations from each antenna to itself. The output of DiFX can then be read by `difx2profile` which will average data from all antennas included in the correlation over the entire length of the correlator job. Due to the required interaction between the VLBA operations staff and the PI, this mode should only be used if the PI is able to commit to short-timescale interaction with the VLBA staff during the correlation stage.

The output of `difx2profile` is an ASCII text file containing two columns of numbers: the bin number and the relative amplitude. In the case of the VLBA, the incoherent sum of ten 25m antennas has the effective sensitivity of a 44m single dish, so profiles of high signal to noise ratio may be difficult to obtain for weak pulsars.

2 Instructions for the PI

The PI of any pulsar project will need to submit to VLBA operations two additional files for correlator configuration. These files are not required for the observation but are needed shortly thereafter. **It is strongly suggested that these files be received by the VLBA analysts no later than 1 week after a pulsar observation.** Note that a separate pair of files is required for each pulsar of a multiple-pulsar observation.

2.1 Pulsar ephemeris file

The ephemeris file is used to predict the pulse phase of a pulsar; it is also called a `.polyco` file. The program `tempo` [1] should be used by the PI to generate the polyco. Currently `tempo` (version 1) is well supported and `tempo2` is only supported in `tempo1` compatibility mode. Eventual support for the `tempo2` *predictors* will be added. All ephemerides must be made for the virtual Earth Center observatory (i.e., XYZ coordinates 0,0,0, usually observatory code 0). Any reference frequency can be specified as the correlator takes dispersion into consideration. Please see the `tempo` documentation for details on the ephemeris file format and instructions on generating the ephemerides.

It is suggested that the ephemeris file adhere to the following naming convention: *experiment*.*pulsar*.`polyco` . For concrete examples, the following would be legal filenames:

- `bb269a.B0834+06.polyco`
- `bz91.crab.polyco`
- `gc045.J1201-3421.polyco`

The pulsar name used in these files should match the pulsar name used when scheduling the observation if possible.

Beware of using `tempo2` ephemeris parameter files with `tempo` (version 1) to produce `.polyco` files as this results in incorrect prediction. Also, as noted above, be aware of the pulse template used when generating the ephemeris and set the gate parameters accordingly if you plan on employing pulsar gating.

2.2 Bin configuration file

All three pulsar modes also require the preparation of a `.binconfig` file. Three pieces of information are contained within this file: the name of the pulsar ephemeris file, definitions of the pulsar bins, and a boolean flag (`SCRUNCH OUTPUT`) that determines whether the bins are weighted and added within the correlator or not. The file consists of a set of keywords (including a colon at the end) that must be space padded to fill the first 20 columns of the file and the values to assign to these keywords that start at column 21. Note that no tabs should appear before the values; the padding to column 21 must make use of only spaces. The file is case sensitive. The pulsar bins each consist of a ending phase and a weight; each bin is implicitly assumed to start when the previous ends and the first bin starts at the end phase of the last. The phases are represented by a value between 0 and 1 and each successive bin must have a larger ending phase than the previous. Examples for some common uses are shown below:

2.2.1 Binary gating

```
NUM POLYCO FILES:   1
POLYCO FILE 0:      BB118A.B0950+08.polyco
NUM PULSAR BINS:    2
SCRUNCH OUTPUT:     TRUE
BIN PHASE END 0:    0.030000
BIN WEIGHT 0:       1.0
BIN PHASE END 1:    0.990000
BIN WEIGHT 1:       0.0
```

2.2.2 Matched-filter gating

```
NUM POLYCO FILES:   1
POLYCO FILE 0:      BB118A.B0950+08.polyco
NUM PULSAR BINS:    6
SCRUNCH OUTPUT:     TRUE
BIN PHASE END 0:    0.010000
BIN WEIGHT 0:       1.0
BIN PHASE END 1:    0.030000
BIN WEIGHT 1:       0.62
BIN PHASE END 2:    0.050000
BIN WEIGHT 2:       0.21
BIN PHASE END 3:    0.950000
BIN WEIGHT 3:       0.0
BIN PHASE END 4:    0.970000
BIN WEIGHT 4:       0.12
BIN PHASE END 5:    0.990000
BIN WEIGHT 5:       0.34
```

Note here that there is zero weight given to pulse phases ranging between 0.05 and 0.95.

2.2.3 Pulsar binning

```
NUM POLYCO FILES:   1
POLYCO FILE 0:      BB118A.B0950+08.polyco
NUM PULSAR BINS:    20
SCRUNCH OUTPUT:     FALSE
BIN PHASE END 0:    0.025000
BIN WEIGHT 0:       1.0
BIN PHASE END 1:    0.075000
BIN WEIGHT 1:       1.0
BIN PHASE END 2:    0.125000
BIN WEIGHT 2:       1.0
.
.
.
BIN PHASE END 18:   0.925000
BIN WEIGHT 18:      1.0
BIN PHASE END 19:   0.975000
BIN WEIGHT 19:      1.0
```

The primary difference is `SCRUNCH OUTPUT: FALSE` which causes each pulsar bin to be written to disk.

It is suggested that the bin configuration file adhere to the following naming convention: *experiment*.*pulsar*.`binconfig` . For concrete examples, the following would be legal filenames:

- `bb269a.B0834+06.binconfig`
- `bz91.crab.binconfig`
- `gc045.J1201-3421.binconfig`

The pulsar name used in these files should match the pulsar name used when scheduling the observation if possible.

3 Instructions for the analysts

The instructions below will typically be used by the VLBA operations staff. Advanced pulsar observers may identify some additional flexibility here or in the `vex2difx` documentation that may be of use.

3.1 Preparing correlator jobs

This guide assumes that `vex2difx` [2] is being used to prepare correlator jobs. `vex2difx` takes as input a `.v2d` file that points to the `.vex.obs` file used (like the `.vex` file used for observing, but with additional data appended) and optionally some additional configuration information. For pulsar observations, one must associate each pulsar with a setup of its own that includes reference to the relevant `.binconfig` file. An excerpt from a `.v2d` file is below:

```
SETUP gateB0950+08
{
        tInt = 2.000
        nChan = 32
        doPolar = True
        binConfig = BB118A.B0950+08.binconfig
}

RULE B0950+08
{
        source = B0950+08
        setup = gateB0950+08
}
```

The `.binconfig` file should be in the same path as the `.v2d` file when running `vex2difx`.

3.2 Making FITS files using `makefits`

The script `makefits` is a wrapper to `difx2fits` used at NRAO to simplify file conversion and to allow an additional level of sanity checking. This script should run just as usually and should automatically make individual FITS files for each pulsar bin.

3.3 Making FITS files using `difx2fits` directly

For the two gating modes, preparing FITS files with `difx2fits` is no different than for any other DiFX output. FITS-IDI does not support multiple phase bins so the pulsar binning case is different and the situation is non-optimal. Each pulsar bin must be made into its own `FITS` file with a separate execution of `difx2fits`. The `-B` (or `--bin`) command line option takes the bin number (starting at zero as above) and writes a FITS file containing data only associated with that bin number. Be sure to systematically name output files such that the bin number is understood.

At NRAO, this direct use of `difx2fits` is limited to testing.

3.4 Profile mode

To make use of `profile` mode, a line `mode=profile` must be placed in the global scope of the `.v2d` file. Further, only a single source can be specified. Pulsar configuration files for this source appropriate for pulsar binning should be created. Instead of forming a FITS file with the output, `difx2profile` should be run as follows:

1. Change directory to `$DIFX_QUEUE/`*project*

2. Run `difx2profile` *filename*`.input`

The output should be a pulse profile file. The weights from this file can be used to set weights in the bin configuration file for a matched-filter gating correlation. Iteration with the PI will be required after this stage before forming the final FITS files.

The power user may be interested in a utility called `profile2binconfig.py`, which, as the name suggests, can create a `.binconfig` file from the output of `difx2profile`.

References

[1] *Tempo*, http://www.atnf.csiro.au/research/pulsar/tempo/

[2] *vex2difx*, http://cira.ivec.org/dokuwiki/doku.php/difx/vex2difx