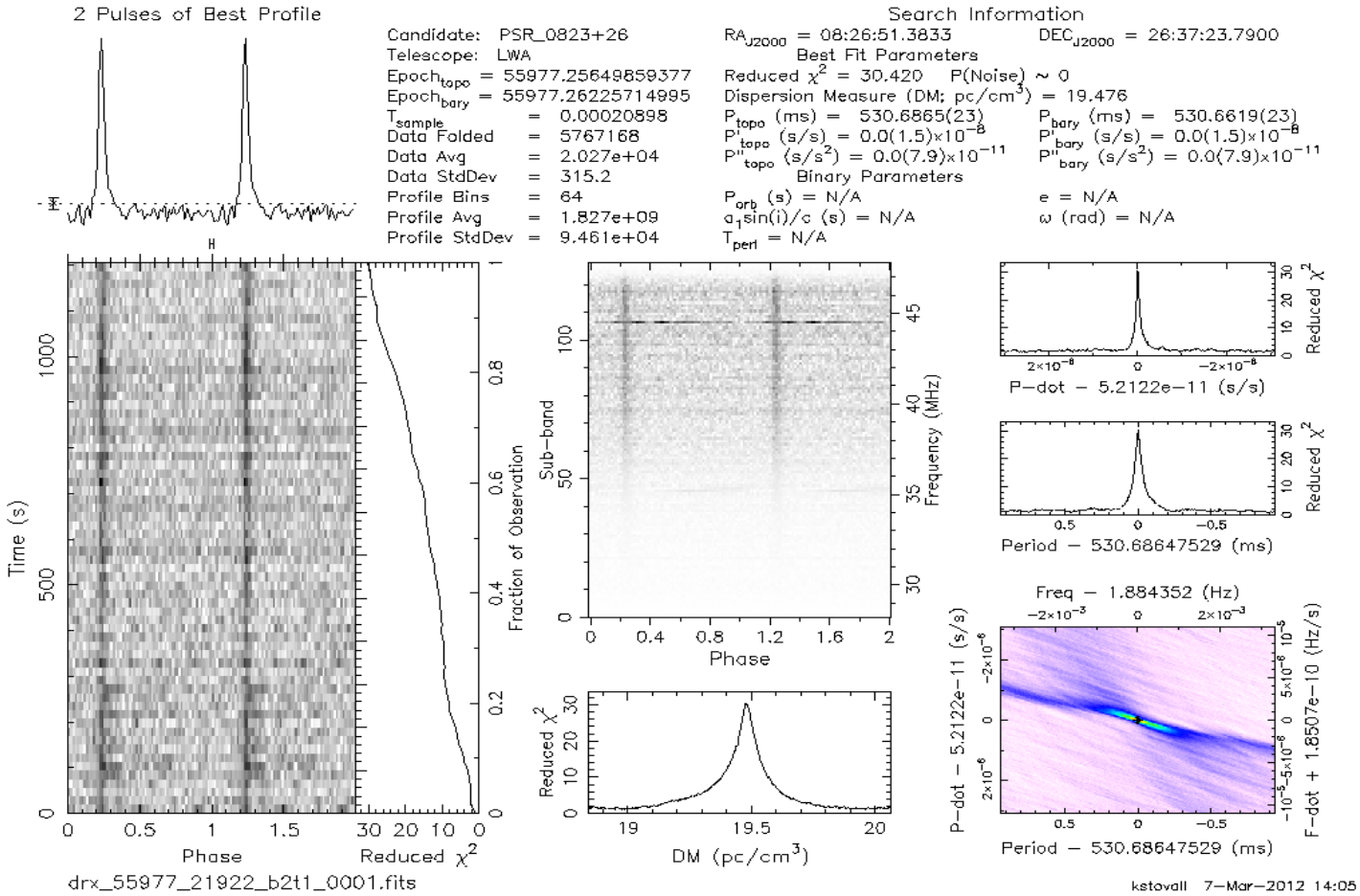
LWA Users Meeting 2012

Pulsars II: Software and Survey

Kevin Stovall



- Pulsar Software
  - PRESTO
  - PSRCHIVE
  - TEMPO/TEMPO2
  - writePsrfits.py
  - Coherent Dedispersion Status

- Pulsar/Transient Survey
  - All Sky Survey
  - LNCC (LWA North Celestial Cap) Survey

PRESTO[1]

- PRESTO is a suite of pulsar search and analysis tools.
- Originally designed to efficiently detect millisecond pulsars in binary systems.
- Its capabilities include RFI detection and excision, incoherent dedispersion, barycentering, acceleration searches, single pulse detection, candidate optimization (folding), TOA generation, plus many others.
- Used in most current pulsar searches.

1) http://www.cv.nrao.edu/~sransom/presto/

PSRCHIVE[1]

- PSRCHIVE is a set of tools used to manipulate folded data sets.
- Designed to provide standard pulsar analysis techniques to a wide range of pulsar data.
- Its capabilities include various plotting tools, interactive RFI zapping, polarimetry and calibration, TOA generation, etc.
- In general, once a pulsar has a “good” ephemeris, much of the analysis is done using PSRCHIVE. Particularly where high precision pulsar timing is necessary.

1) http://psrchive.sourceforge.net/

TEMPO/TEMPO2

- TEMPO and TEMPO2 are used to analyze TOAs, compare TOAs to an ephemeris, and to find an ephemeris which better fits the TOAs.
- Compares TOAs to a variety of model parameters including astrometric, frequency and frequency derivatives, and binary parameters.
- In the majority of pulsar analysis, TOAs from PRESTO or PSRCHIVE are compared with the pulsar's model using either TEMPO or TEMPO2.

writePsrfits.py

- PSRFITS[1] is a standard pulsar data storage format that is in use by most of the current pulsar recording instruments (also supported by both PRESTO and PSRCHIVE).

- writePsrfits.py takes LWA DRX format and converts it into a filterbank stored in PSRFITS format.

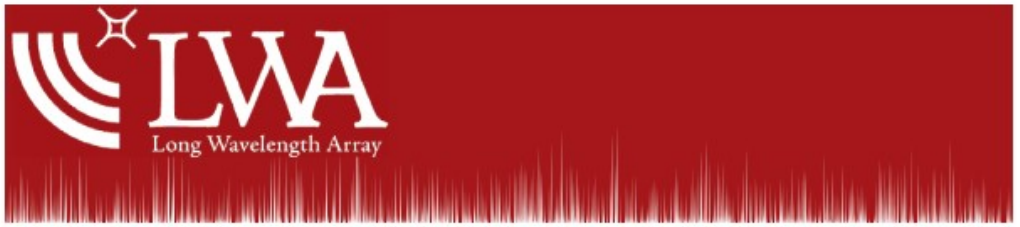
  - Uses existing LSL (LWA Software Library) modules to read in a small amount of DRX data and calculate a power spectrum, then uses psrfits_utils modules to write the spectra in PSRFITS format.

  - Minor tweaks of PRESTO, PSRCHIVE, and TEMPO are required make our output compatible with the output of writePsrfits.py.

1) http://www.atnf.csiro.au/research/pulsar/psrfits/index.html

writePsrfits.py

http://fornax.phys.unm.edu/lwa/trac/wiki/Pulsar



Pulsar Tools for LWA-1

Table of Contents

Overview and Requirements

The pulsar tools build off the LSL 0.4.x framework for working with LWA data and also depends on the following python modules:

- cfitsio
- psrfits_utils (available at https://github.com/kstovall/psrfits_utils)

No installation (e.g., python setup.py install) is required to use the software. Simply run the scripts in the Pulsar directory.

Obtaining

The easiest way to obtain the latest version of the pulsar tools is via subversion export/checkout:

```
svn checkout http://fornax.phys.unm.edu/lwa/subversion/trunk/Pulsar
```

Contents

writePsrfits.py

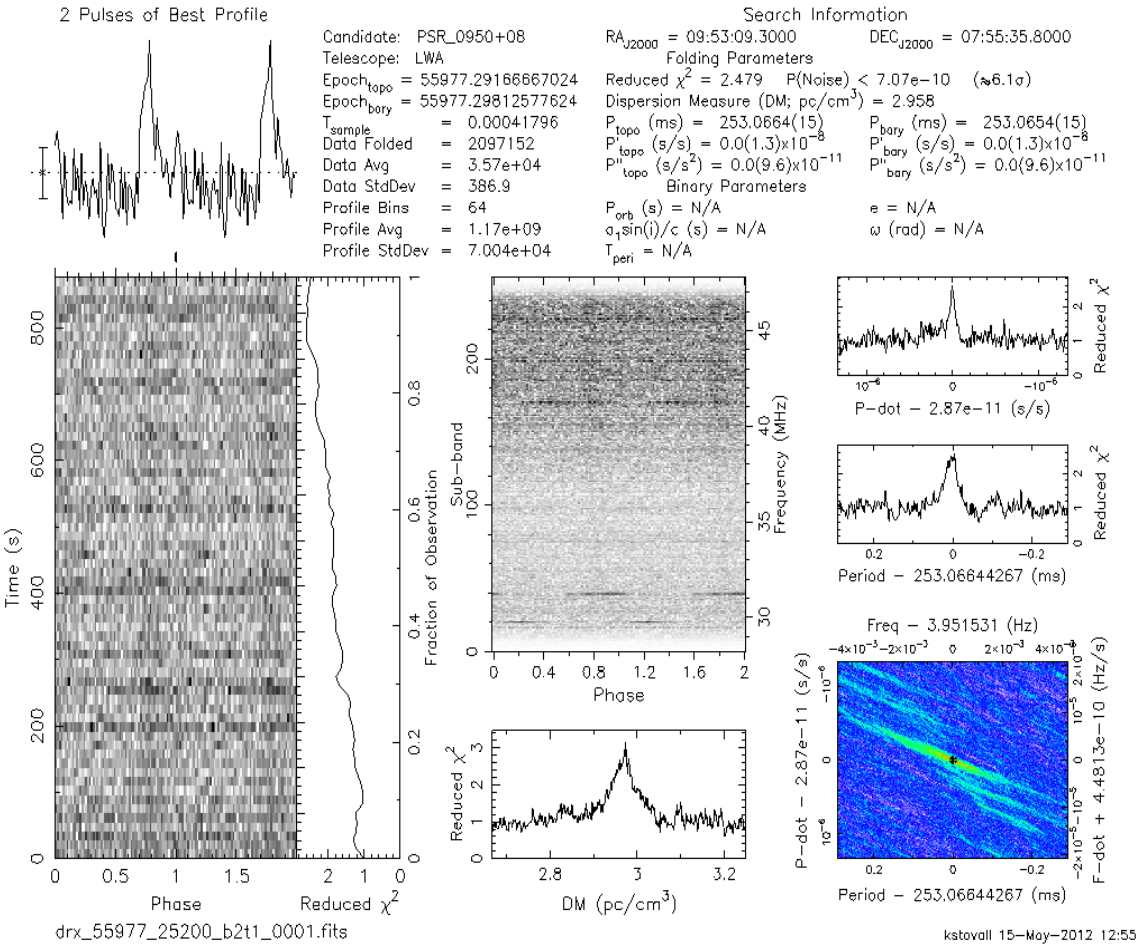
Given a RA, Dec., and DRX file, create a PSR FITS file of the data.

```
writePsrfits.py --ra=HH:MM:SS.SS --dec=DD:MM:SS.SS drxfilename
```

writePsrfits.py

Additions to writePsrfits.py in the near future:

- More flexible parameters
  - Default number of frequency channels is 4096, generalized code written, but it is not yet in the LSL repository.
- More accurate header information
- Full Stokes Parameters (This with fold_psrfits will enable us to do polarization analysis using the LWA).



Coherent Dedispersion

$$V_{coh}(f_0+f)=V(f_0+f)H^{-1}(f_0+f)$$

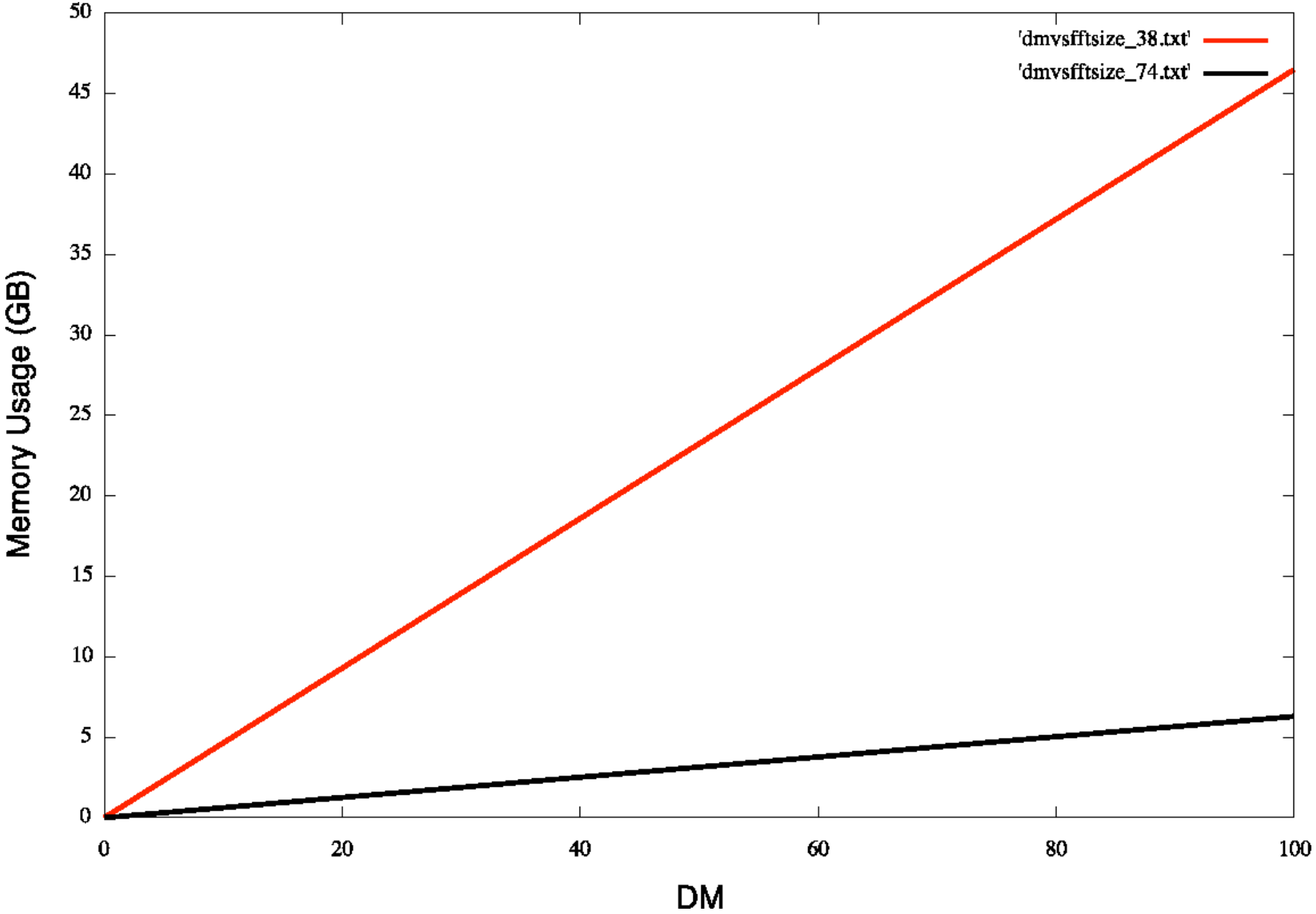
$$H(f_0+f)=e^{i\frac{2\pi D}{(f+f_0)f_0^2}DM\ f^2}$$

$$D=\frac{e^2}{2\pi m_e c}=4148.808(3)\frac{MHz^2\,cm^3\,s}{pc}$$

$$\Delta t\sim 4.15\times 10^6\,ms\times(f_1^{-2}-f_2^{-2})\times DM$$

A pulsar with a DM of 5 observed at 38 MHz with 19.6MHz bandwidth requires a minimum of ~300,000,000 point fft.

Coherent Dedispersion



Coherent Dedispersion

Two possible routes:

- DSPSR[1]
  - DSPSR is the software used by most pulsar backends that support coherent dedispersion.
  - We would need to add the DRX format to the backends supported by DSPSR.
  - DSPSR may have problems at our frequencies unless we make some changes to the main dedispersion code.

- Write our own
  - Use existing DRX reader code in LSL.
  - May be easier to scale across multiple nodes?

1) http://dspsr.sourceforge.net/

All Sky Pulsar/Transient Survey

- Using a single beam with tunings centered at 38MHz and 54MHz each with 19.6MHz bandwidth. The entire LWA visible sky can be covered using ~3,500 pointings.
- Simulations suggest that 1 hour dwells with this setup will allow the detection of more than 500 pulsars, with a few tens of them being new discoveries.
- The data will be searched for transients as well. Depending on the RRAT population, we may be able to discover a few hundred new ones.

LWA Northern Celestial Cap (LNCC) Survey

- For the LWA CFP2, we submitted a proposal to begin the initial stages of the LWA All Sky Pulsar/Transient Survey by using 320 beam hours to survey the area of the sky north of 54 degrees.

- About 75 pulsars should be detected in the LNCC.

- A few tens of RRATs should also be detectable.

LWA Northern Celestial Cap (LNCC) Survey

- Data would be searched using a PRESTO based search pipeline modeled after pipelines used in current pulsar/transient surveys at the GBT and Arecibo.
- Processing is planned to be done using clusters at the Texas Advanced Computing Center and at the University of Texas at Brownsville.
- The 320 beams proposed would result in about 16,000 pulsar candidates for human inspection. The inspection will be done by high school and undergraduate students in the Arecibo Remote Command Center (ARCC) program, who have found ~30 new pulsars within the past 1.5 years.

LWA Northern Celestial Cap (LNCC) Survey

The following further software developments are planned in the near future in preparation for the LNCC survey:

- Tool to combine two tunings which are spaced appropriately in frequency into a single observation.
- WritePsrfits.py will need to be modified to be able to downsample prior to outputting PSRFITS file.