Introduction to X-ray Pulsar Timing with PINT

NICER/IXPE Workshop, GWU
2024 July 31

Megan DeCesar
George Mason University

General introduction to pulsar timing

Pulsar timing models

- Phase connected or coherent timing model: accounts for every pulse cycle, and can thus predict the time of every pulse arriving at the telescope

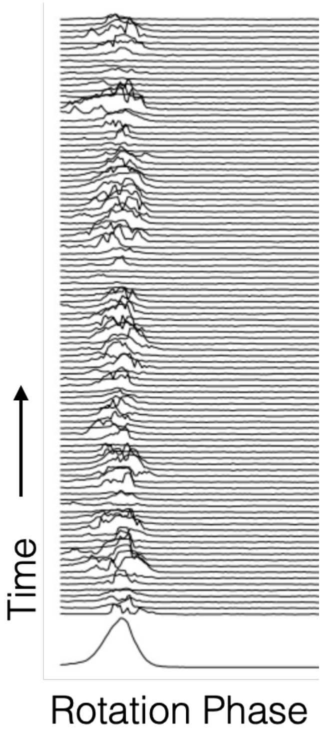
- Timing model is built by accounting for every observed pulse time of arrival (TOA)

- Model parameters that can be determined include:
  - Spin: frequency (ν), spindown rate ($d\nu/dt$), additional frequency derivatives
  - Orbital (P_{orb}, T_0, e, ω, $x = a \sin i$, post-Keplerian or GR terms)
  - Astrometric (position $\{\alpha, \delta\}$, parallax π, proper motion μ)

Note: Sometimes event times are referred to as TOAs; they are not the same thing as pulse TOAs!

"Folding" to produce a stable pulse profile

Every pulse is different -> must "fold" and measure arrival time of summed profile



This is a radio example.

With X-rays / high-E events, we might record events from only a few phases per rotation.

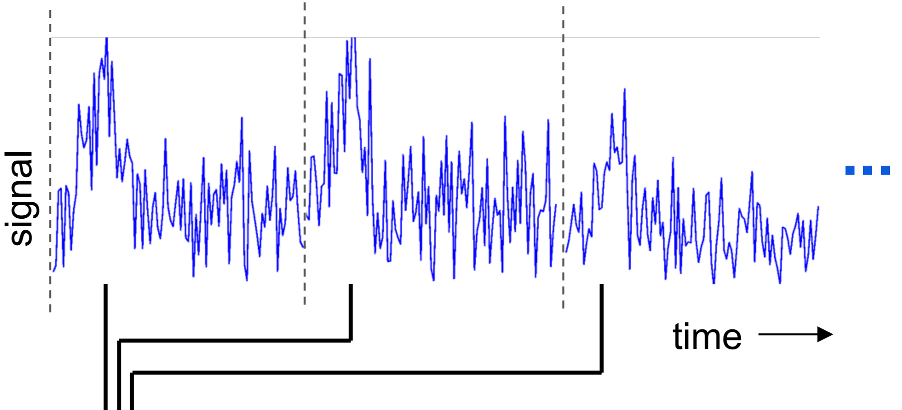
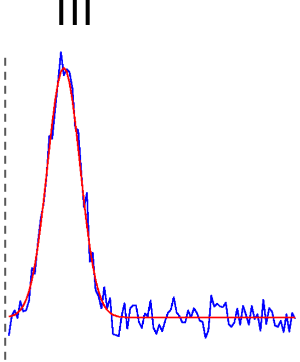


Figure: David Nice, Paul Ray

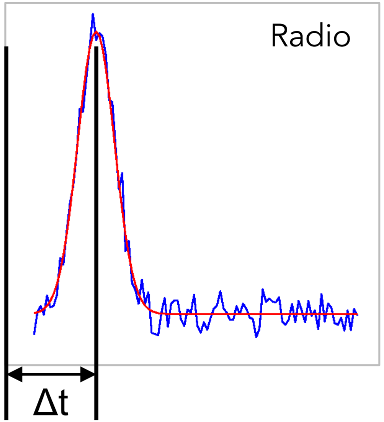


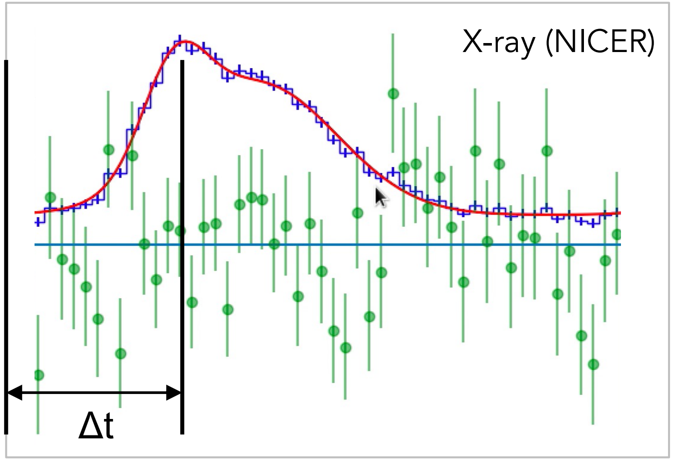
Fold 10^2 - 10^4 pulses to produce pulse profile that is stable over long timescales

Measuring Pulse Times of Arrival (TOAs)

The data are pulse times of arrival (TOAs) at the telescope:

- All TOAs for a given timing analysis are relative to same fiducial time T_0
- Make a profile template; shift template to fit the observed profile to measure ΔT
- TOA precision depends on template fit, width of pulse (narrower pulse $\rightarrow$ higher precision)
- Note that <100 ns event time precision $\not\rightarrow$ TOA precision of 100 ns





Pulse Time of Arrival:
TOA = scan start time + ΔT

Measured at ground-based (radio) observatory: ***topocentric*** TOAs

We also refer to TOAs at the spacecraft as topocentric.

The "timfile"

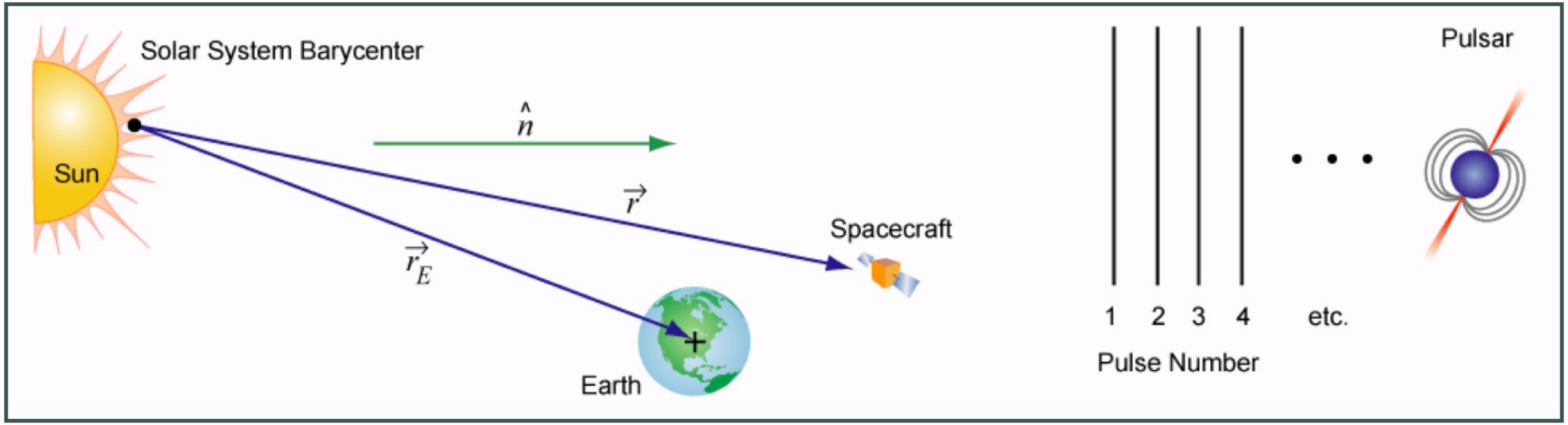
Example of TOA lines from a timing model file, with standard naming convention PSRNAME.tim

- "Observatory" can be ground observatory for radio TOAs
- For high-energy TOAs, it can be the space-based observatory, the solar system barycenter, or the geocenter (Earth's center of mass; often used by *Fermi*)

Figure: Paul Ray

Observatory			Radio Frequency (MHz)	Pulse Time of Arrival (MJD)	Measurement Uncertainty (μs)		
a	3751	1518+49	370.000	50942.02369981804596	69.1	9-May-98	460.2
a	3751	1518+49	370.000	50942.02508871578912	74.9	9-May-98	460.8
a	3752	1518+49	370.000	50942.02710263928441	107.8	9-May-98	460.1
a	3752	1518+49	370.000	50942.02849153928888	68.4	9-May-98	463.7
a	3753	1518+49	370.000	50942.03050309034722	63.0	9-May-98	459.7
a	3753	1518+49	370.000	50942.03189199466585	71.4	9-May-98	468.7
a	3754	1518+49	370.000	50942.03389643284537	64.2	9-May-98	461.5
a	3754	1518+49	370.000	50942.03528532340819	57.4	9-May-98	456.3
a	3755	1518+49	370.000	50942.03728740139970	74.4	9-May-98	459.7
a	3755	1518+49	370.000	50942.03867629785610	65.1	9-May-98	461.5
a	3756	1518+49	370.000	50942.04067884384616	54.2	9-May-98	458.8
a	3756	1518+49	370.000	50942.04206774860490	87.3	9-May-98	470.2
a	3757	1518+49	370.000	50942.04406981298474	88.9	9-May-98	461.2
a	3757	1518+49	370.000	50942.04545870833792	71.8	9-May-98	463.1
a	3758	1518+49	370.000	50942.04748447411745	110.3	9-May-98	463.0
a	3758	1518+49	370.000	50942.04887336536594	78.6	9-May-98	461.1
a	3759	1518+49	370.000	50942.05089865820880	60.2	9-May-98	463.4
a	3759	1518+49	370.000	50942.05228755033977	131.1	9-May-98	463.1
a	3760	1518+49	370.000	50942.05428961858992	63.4	9-May-98	460.9
a	3760	1518+49	370.000	50942.05567851214494	93.2	9-May-98	462.8
a	3761	1518+49	370.000	50942.05768105475176	116.2	9-May-98	461.0
a	3761	1518+49	370.000	50942.05906994776154	75.0	9-May-98	463.0
a	3762	1518+49	370.000	50942.06108244410689	72.2	9-May-98	465.9
a	3762	1518+49	370.000	50942.06247133259781	76.9	9-May-98	463.6
a	3763	1518+49	370.000	50942.06450988581265	86.1	9-May-98	461.4
a	3763	1518+49	370.000	50942.06589877480622	61.9	9-May-98	460.4
a	3764	1518+49	370.000	50942.06790794988299	90.1	9-May-98	460.5
a	3764	1518+49	370.000	50942.06929683956486	67.2	9-May-98	460.8
a	3765	1518+49	370.000	50942.07129227137214	63.5	9-May-98	460.6
a	3765	1518+49	370.000	50942.07268116130441	139.5	9-May-98	461.8

Barycentering TOAs



- Topocentric TOAs must be converted to a nearby inertial frame, typically the Solar System barycenter, before fitting the timing model
- This removes effects of observer motion and relativistic clock effects

Fitting TOAs to the timing model

The goal is to find model parameters that minimize residuals between data (measured TOAs) and model (predicted TOAs)

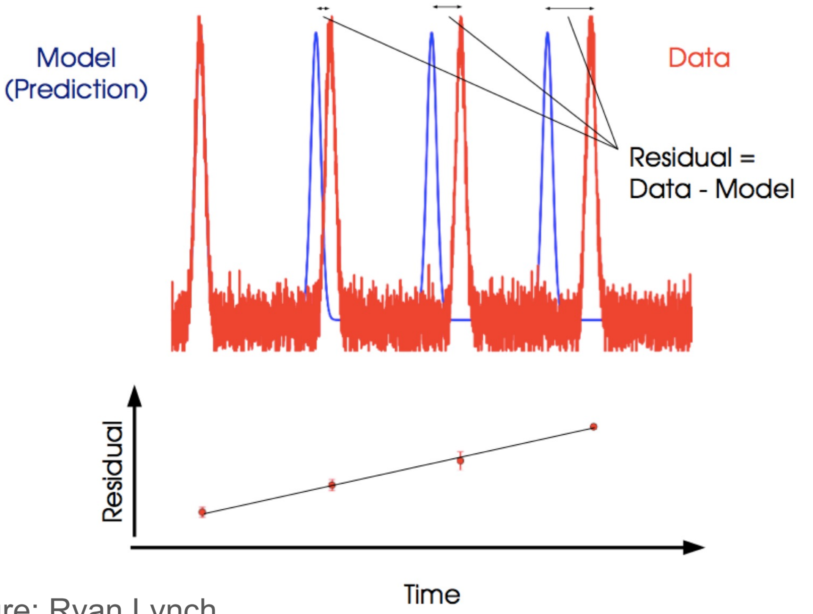


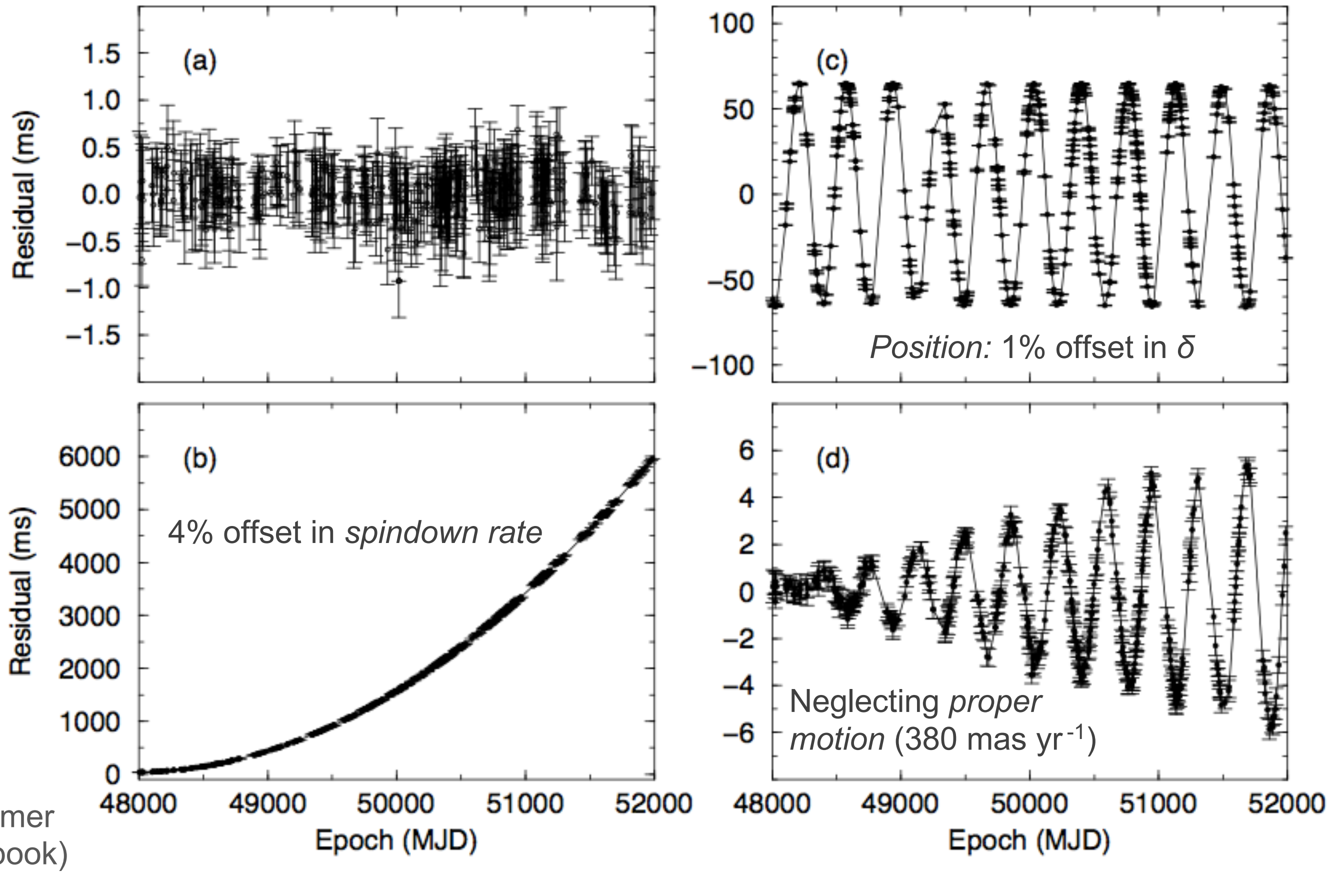
Figure: Ryan Lynch

Very simple model: Only need to fit for spin frequency

Fitting TOAs to the timing model

Example residuals

- Using radio data
- Long timing baseline
- Individual parameters perturbed from best-fit values

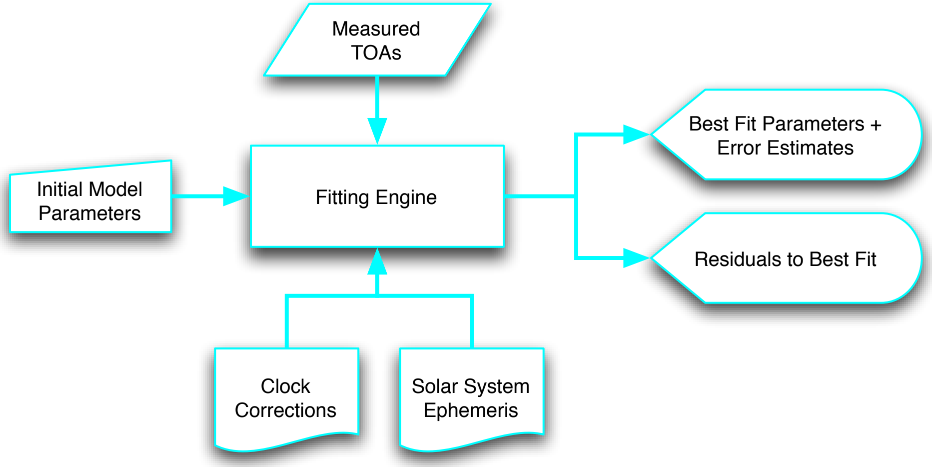


Lorimer & Kramer (Pulsar Handbook)

Fitting TOAs to the timing model (parfile)

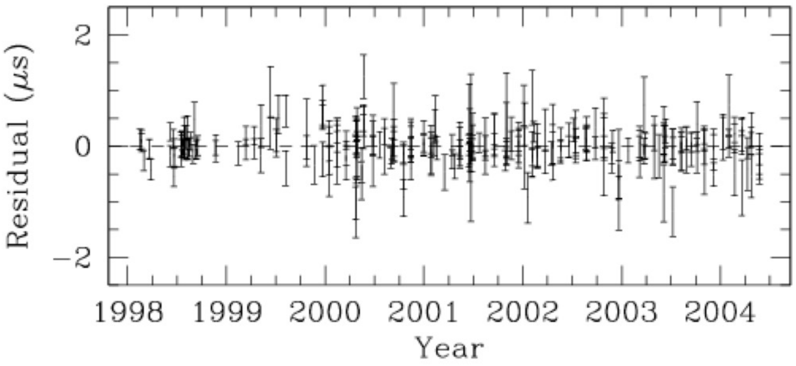
Example parfile

```
PSRJ           J1513-5908
RAJ            15:13:55.6713462
DECJ           -59:08:09.16532
F0             6.5970918979437165796
F1             -6.6530805250943988541e-11
PEPOCH         55336
POSEPOCH       54710
START          54220.401890338307769      1
FINISH         58469.808981595782623      1
TZRMJD         57675.99293714394800
TZRSITE        pks
TRES           1054.132
EPHVER         2
CLK            TT(TAI)
UNITS          TDB
```

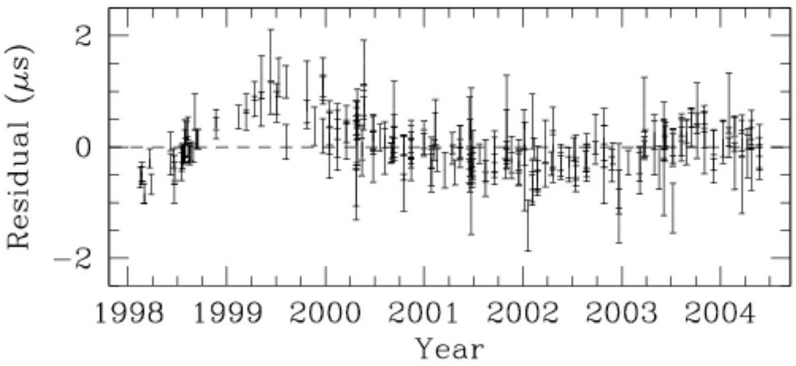


$$\phi(t) = \phi(0) + \nu t + \frac{1}{2}\dot{\nu}t^2 + \frac{1}{6}\ddot{\nu}t^3 + \ldots$$

Effects of choice of Solar System ephemeris



PSR J1713+0747 analyzed using DE405 solar system ephemeris



PSR J1713+0747 analyzed using previous-generation DE200 solar system ephemeris.

~1μs timing errors ⇔ 300 m errors in Earth position.

Splaver+ 2005

Fitting TOAs to the timing model

- Several software options
- If you are starting out with timing, recommend learning PINT straight away

TOOLS FOR FITTING TIMING MODELS

- Tempo <http://tempo.sourceforge.net/>
  - Developed by Princeton and ATNF over 30+ years
  - Well tested and heavily used
  - Based on TDB time system
  - But, nearly undocumented, archaic FORTRAN code
- Tempo2 < https://bitbucket.org/psrsoft/tempo2>
  - Developed at ATNF recently
  - Based on TCB time system (coordinate time based on SI second)
  - Better documented, modern C code, uses long double (128 bit) throughout
  - Useful plug-in architecture to extend capabilities
- PINT <https://github.com/nanograv/PINT>
  - Modern, modular Python code for pulsar timing
  - Heavy use of well-debugged libraries (IAU SOFA, astropy)
  - Code independent of Tempo/Tempo2

Time Systems

TAI = Atomic time based on the SI second
UT1 = Time based on rotation of the Earth
UTC = TAI + "leap seconds" to stay close to UT1
TT = TAI + 32.184 s
TDB = TT + periodic terms to be uniform at SSB
TCB = Coordinate time at SSB, based on SI second

Figure: Paul Ray

X-ray pulsar timing with NICER and PINT

Software used in this tutorial

HEASARC ftools:
https://heasarc.gsfc.nasa.gov/ftools/

- nicerl2 (NICER ftool)
- barycorr
- niextlc
- niextract-events

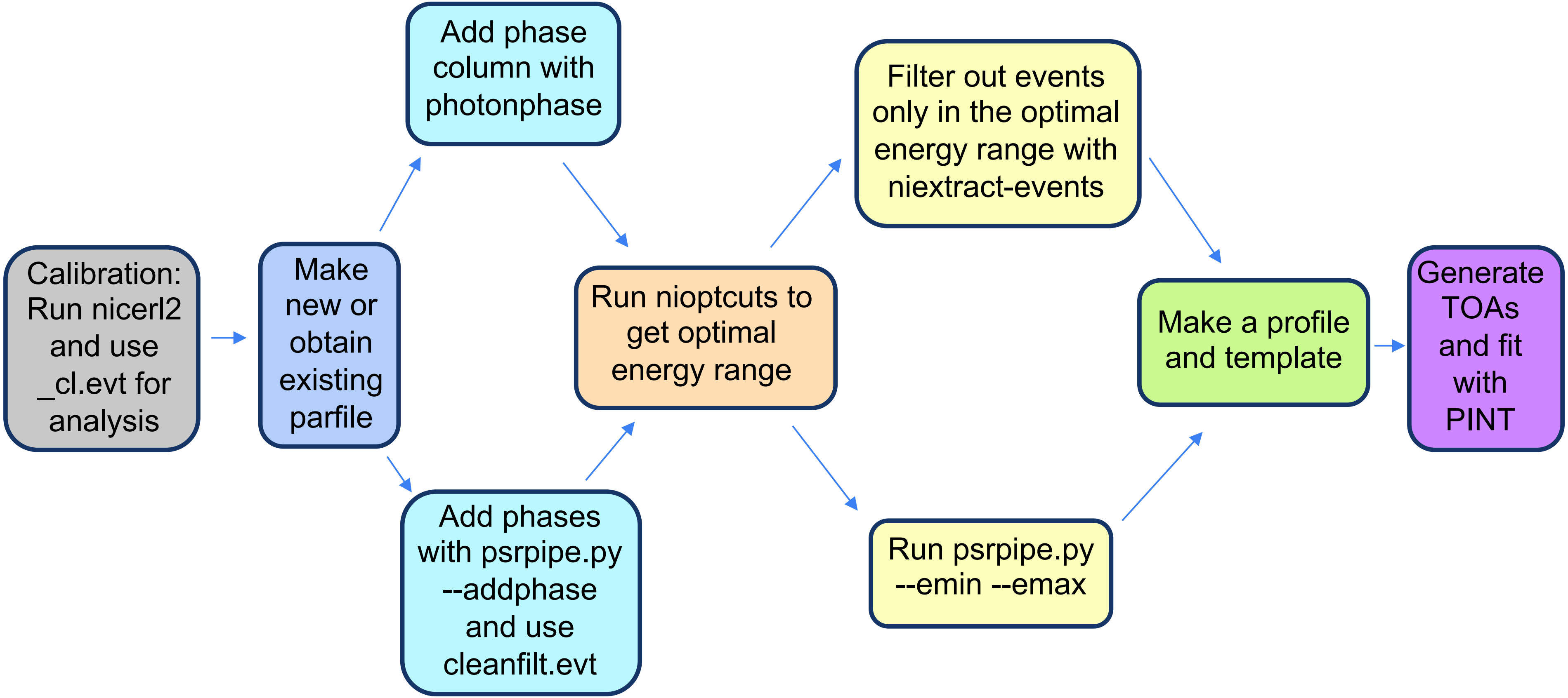
PINT: https://github.com/nanograv/PINT

- photonphase
- pintk

NICERsoft:
https://github.com/paulray/NICERsoft

- A repository with user-contributed tools
- Not official software (not distributed by HEASARC)
- psrpipe.py
- cr_cut.py
- photon_toa.py
- remove_empty_evtfiles.py

Summary of analysis steps



Calibrate and make clean event file with nicerl2

```
> nicerl2 indir=<source>/<obsid> incremental=NO tasks=ALL
filtcolumns=NICERV5 clobber=YES
```

This produces a file with filename ending in"_cl.evt"

➡ This is the clean, calibrated file to use for analysis.

Q: Can I ever just start from _cl.evt in the downloaded dataset?
A: The recommendation is to always run nicerl2 before proceeding with other analyses.

Make or obtain a timing model (parfile)

Making a parfile for a pulsar without an existing timing model: You can start with fitting just the spin frequency, F0.

E.g.:

PSR J1234+56
RAJ 12:34:00
DECJ +56:00:00
F0 0.01234 1
EPHEM DE405
CLK TT(TAI)
UNITS TDB

Some software resources: Stingray, HENDRICS, psrfits2presto

Using an existing parfile

- For a known pulsar, you can likely find an existing parfile, e.g. in a paper that previously timed the pulsar in question.
- Note: Existing parfiles can be out of date. You may be able to calculate reliable pulse phases for only the part of your data that overlaps in time with the
- A useful resource is the *Fermi* LAT 3rd Pulsar Catalog individual pulsar summaries page.

Compute phases and add PULSE_PHASE column to _cl.evt

Option 1: First run the ftool barycorr[1,2] --refframe=ICRS[3] on the *_cl.evt file to barycenter the event times, and then run PINT's photonphase without the "--orbfile" option

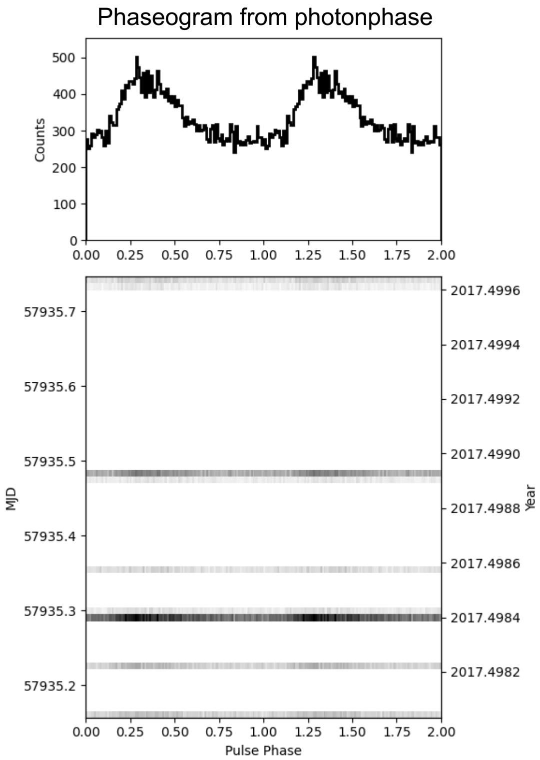
```
> barycorr infile=0020020108/xti/event_cl/ni0020020108_0mpu7_cl.evt
outfile=0020020108/xti/event_cl/ni0020020108_0mpu7_cl_bary.evt
orbitfiles=0020020108/auxil/ni0020020108.orb refframe=ICRS

> photonphase --addphase --ephem DE405
--outfile 0020020108/xti/event_cl/ni0020020108_0mpu7_cl_bary_ph.evt
--plot --plotfile 0020020108/xti/event_cl/B1509_0020020108_phaseogram.png
0020020108/xti/event_cl/ni0020020108_0mpu7_cl_bary.evt J1513-5908_PKS.par
```

[1]Must specify ra= and dec= if NICER pointing position wasn't directly toward your pulsar.
[2]barycorr does not work for all missions; e.g., Fermi uses gtbary.
[3]refframe=ICRS is needed in order to use SSEs that are more recent than DE200 (e.g. DE405).

Compute phases and add PULSE_PHASE column to _cl.evt

Option 2: Starting from the *_cl.evt file, run the PINT command **photonphase**[4,5], including the **--orbfile** option to compute barycentered phases

```
> photonphase --addphase --ephem DE405 --plot
--plotfile 0020020108/xti/event_cl/B1509_0020020108_phaseogram.png
--orbfile 0020020108/auxil/ni0020020108.orb
--outfile 0020020108/xti/event_cl/ni0020020108_0mpu7_cl_ph.evt
0020020108/xti/event_cl/ni0020020108_0mpu7_cl.evt
J1513-5908_PKS.par
```



[4]**photonphase** works for many but not all missions. Some require the event file to be barycentered first. Fermi has a specific tool, **fermiphase**.

[5]**photonphase** can be pretty slow. For large numbers of events, first barycenter your data, and then run **photonphase --polycos** to speed it up.

NOTE: (a) To run on multiple files, must loop through them. (b) Can also make phaseogram later with **niphaseogram**.

Clean data and compute phases with psrpipe.py

Run psrpipe.py, including the parfile

You can just use the default options, so simply run:

```
> psrpipe.py --par <parfile> --ephem <parfile ephem> <raw directory(ies)>
```

Here is the command with defaults implicit, plus some common additional options:

```
> psrpipe.py --cormin 1.5 --kpmax 5 --tidy --nomap --par <parfile> --ephem
<parfile ephem> <raw directory(ies)>
```

Cosmic ray rigidity

Solar KP-index

You will likely need --nomap because of cartopy (a python package) errors

Note: Full command with defaults being input explicitly:

```
> psrpipe.py --emin 0.22 --emax 15.0 --mingti 16 --maxovershoot 1.5
--maxundershoot 200 --medianundershoot 100 --par <parfile> --ephem <parfile
ephem> <raw directory(ies)>
```

Video: Running psrpipe.py

```
(pint) hesenode1%
(pint) hesenode1%
(pint) hesenode1%
(pint) hesenode1%
(pint) hesenode1%
(pint) hesenode1%
```

Output from psrpipe.py (obsIDs with events remaining)

```
(pint) hesenode1% ls
0020020101_pipe  0020020105_pipe  0020020109_pipe  0020020113_pipe  1020020101_pipe  1020020105_pipe  6020020104_pipe
0020020102_pipe  0020020106_pipe  0020020110_pipe  0020020114_pipe  1020020102_pipe  6020020101_pipe  6020020105_pipe
0020020103_pipe  0020020107_pipe  0020020111_pipe  0020020115_pipe  1020020103_pipe  6020020102_pipe  J1513-5908_PKS.par
0020020104_pipe  0020020108_pipe  0020020112_pipe  0020020116_pipe  1020020104_pipe  6020020103_pipe
(pint) hesenode1%
(pint) hesenode1% ls 0020020108_pipe
0020020108_cleanfilt_bkg.png  0020020108_cleanfilt_sci.png  cleanfilt.lc   cleanfilt.pha      phaseogram.png     tot_and_eventgti.gti  tot.gti
0020020108_cleanfilt_eng.png  cleanfilt.evt                 cleanfilt.mkf  ni0020020108.orb   psrpipe_expr.txt   tot_clipped.gti
```

fv: Summary of cleanfilt.evt in /home/mdecesar/nicer/analysis/PSRB1509-58_timing/my_pipe/0020...

File Edit Tools Help

Index	Extension	Type	Dimension	View
0	Primary	Image	0	Header Image Table
1	EVENTS	Binary	15 cols X 4656 rows	Header Hist Plot All Select
2	GTI	Binary	2 cols X 1 rows	Header Hist Plot All Select
3	FPM_SEL			
4	GTI_MPU0			
5	GTI_MPU1			
6	GTI_MPU2			
7	GTI_MPU3			
8	GTI_MPU4			
9	GTI_MPU5			
10	GTI_MPU6			

fv: Binary Table of cleanfilt.evt[1] in /home/mdecesar/nicer/analysis/PSRB1509-58_timing/my_pipe/0020020106_pipe/

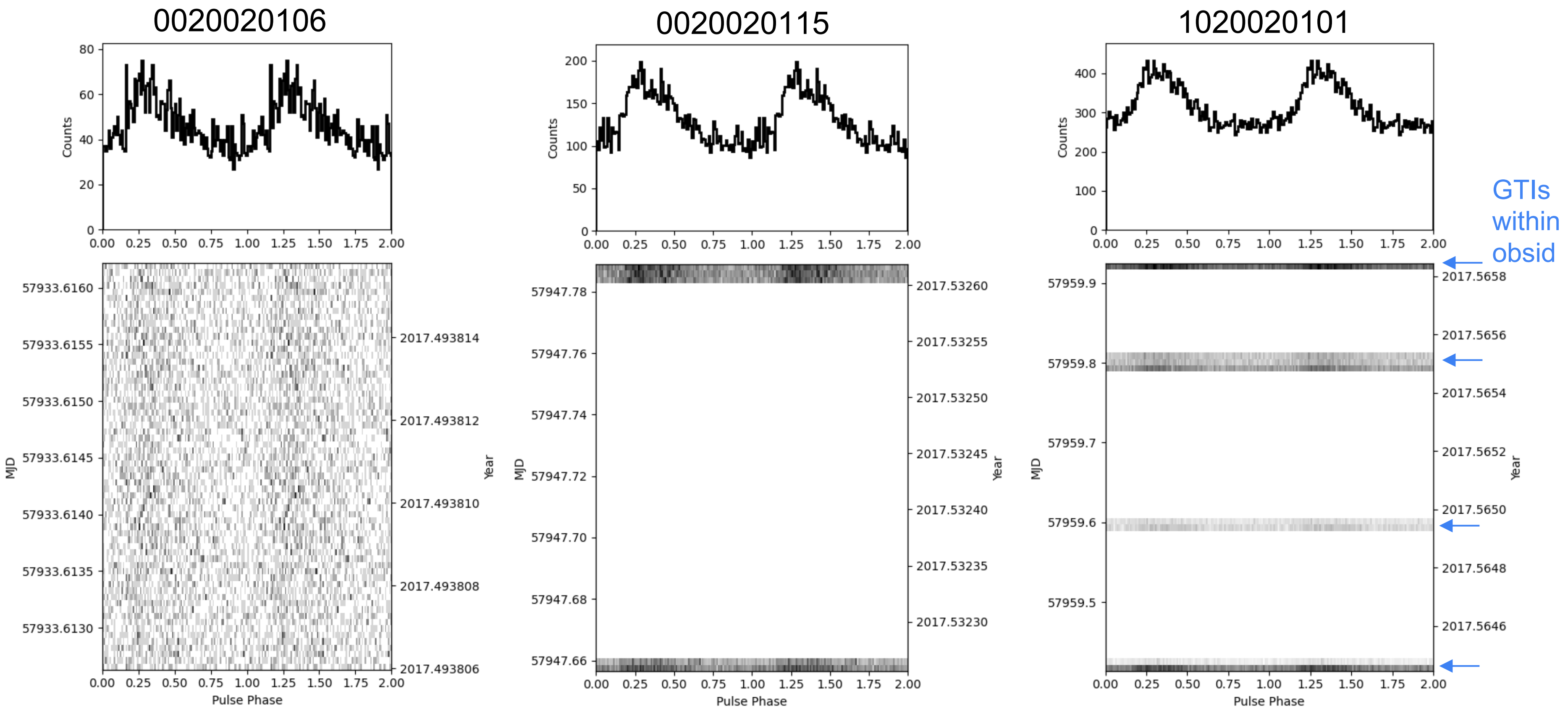
File Edit Tools Help

Select	TIME	RAWX	RAWY	PHA	PHA_FAST	DET_ID	DEADTIME	EVENT_FLAGS	TICK	MPU_A_TEMP	MPU_UNDER_COUNT	PI_FAST	PI	PI_RATIO	PULSE_PHASE
	1D	1B	1B	1I	1I	1B	1B	8X	1K	I	J	1I	1I	1E	D
All	s	pixel	pixel	chan	chan		s			Celsius		chan	chan		
Invert	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify	Modify
1	110212865.133434996	0	0	690	781	57	0.000022637	0	30467848259319	33.155	7	148	142	9.594595E-01	6.175405099342E-01
2	110212865.415276006	0	5	1368	1331	63	0.000022637	0	30465965199829	32.731	6	379	373	9.841689E-01	4.726361481516E-01
3	110212865.546791002	0	1	2531	2439	56	0.000022637	0	30467858925362	33.155	6	867	849	9.792387E-01	3.382767918745E-01
4	110212865.567445993	3	2	1241	1171	44	0.000022637	0	30469654359472	34.109	5	337	337	1.000000E+00	4.742287683582E-01
5	110212865.657436997	4	6	1595	1607	30	0.000022637	0	30472604587731	36.124	3	486	477	9.814815E-01	6.655357269412E-02
6	110212865.662405998	0	3	1258	1233	67	0.000022637	0	30465971576627	32.731	4	344	346	1.005814E+00	9.925940775099E-02
7	110212865.797363997	3	6	815	803	40	0.000022637	0	30469660292158	34.109	10	173	168	9.710982E-01	9.875635162368E-01
8	110212865.823895007	1	4	1018	974	64	0.000022637	0	30465975743613	32.731	5	254	248	9.763780E-01	1.621938095739E-01
9	110212865.929389998	6	1	879	807	15	0.000022637	0	30476652414525	35.275	5	209	201	9.617225E-01	8.565675145559E-01
10	110212865.942524999	3	4	778	734	42	0.000022637	0	30469664037829	34.109	5	160	153	9.562500E-01	9.430254316474E-01
11	110212865.979012996	3	2	936	898	44	0.000022637	0	30469664979332	34.109	5	228	222	9.736842E-01	1.831880361798E-01
12	110212865.988566995	4	4	1480	1403	32	0.000022637	0	30472613132042	36.124	5	423	423	1.000000E+00	2.460747613406E-01
13	110212866.046950996	7	5	913	966	01	0.000022637	0	30479089482364	34.745	4	216	219	1.013889E+00	6.303627289205E-01

Go to: Edit cell:

Event times in MET

Phaseograms and GTIs



Next step is finding optimal energy ranges ... but first a quick aside

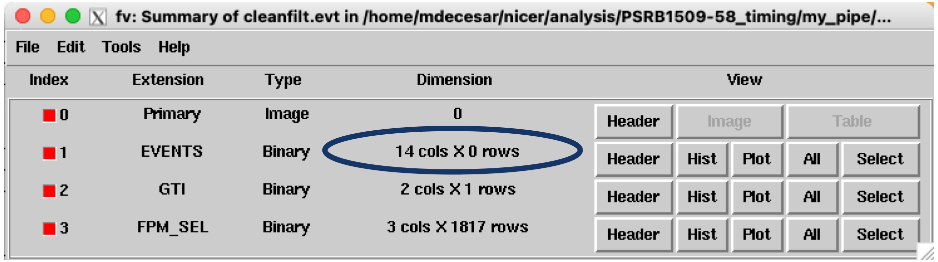
- `psrpipe.py` produced the output file cleanfilt.evt for each obsid

- If you used an already-existing, phase-connected parfile (that is valid over your full dataset) to compute PULSE_PHASE, then you can:
  - Remove files with no remaining data after the `psrpipe.py` filtering step
  - Merge the cleanfilt.evt files into a single merged.evt file
  - *Optional:* Make a light curve (over time – not the same as a pulse profile) from merged.evt; check for and remove flaring; output is merged_cut.evt
  - Use merged.evt or merged_cut.evt for the rest of the analysis

- If instead you have a new parfile, e.g. that has only F0 measured from single obsid, then:
  - You cannot yet merge all the obsids – first need a phase-connected timing model
  - *Optional:* Make light curve and check for flaring (output is cleanfilt_cut.evt)
  - Do the rest of the analysis on individual cleanfilt.evt or cleanfilt_cut.evt files
  - After finding a timing solution using TOAs from the full dataset, you can re-compute PULSE_PHASE with this good solution, and then merge the files as described above

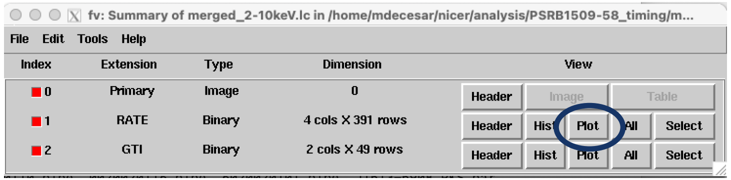
For this tutorial, I'll use a merged event file

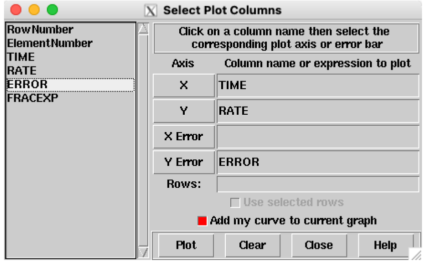
- Remove files with no data and merge
    - Make file list, remove empty files, merge

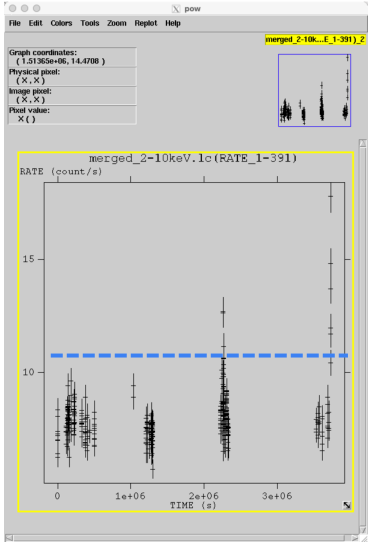



```
> /bin/ls -1 *pipe/cleanfilt.evt > files.txt
> remove_empty_evtfiles.py files.txt cleanfiles.txt
> niextract-events @cleanfiles.txt merged.evt
```

- Optional: Make light curve and examine in "fv" to determine "--cut" value








```
> niextlc merged.evt merged_2-10keV.lc timebin=32
pirange=200:1000 lcthresh=0.9 clobber=yes
> fv merged.evt
> cr_cut.py merged.evt --outname merged_cut.evt --cut
10.6 --filterbinsize=32.0 --lcfile merged_2-10keV.lc
```

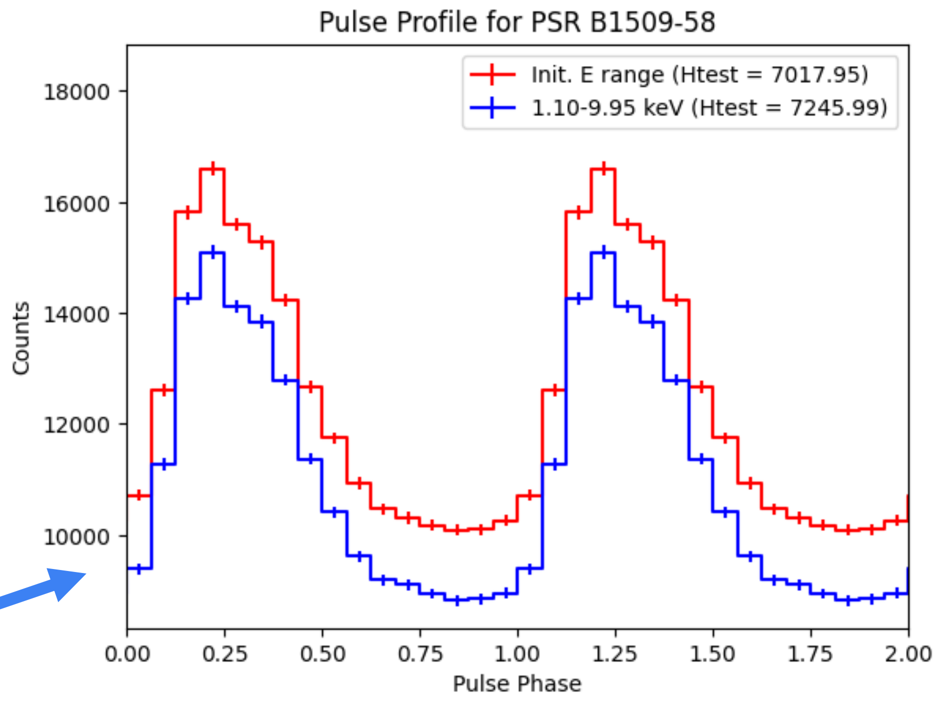
Find optimal energy range for pulsations

```
> nioptcuts.py merged_cut.evt
```

```
(pint) hesenode1% nioptcuts.py merged_cut.evt
 - Reading file merged_cut.evt
 - Opening events for PSR B1509-58
t: At col 0, Unit 'Celsius' not supported by the FITS standard.
f_unit'. To have it recognized inside a file reader or other code
ttps://docs.astropy.org/en/latest/units/combining_and_defining.ht

  - Concatenating files
  - Initial Htest = 7017.95 (82.804 sigma) in range 0.22-15.00 keV
██████████████████████████████████████████████| 7

  - Final Htest = 7245.991 (84.162 sigma)
  - Best range:  emin 1.1 keV -- emax 9.95 keV
```

Make profile in optimal energy band:

```
> RPP-profile.py --optemin 1.1
--optemax 9.95 merged_cut.evt
```



Filter event file to include only the optimal energy range

Make event file with optimal energy range only with niextract-events:

```
> niextract-events
  Input file name:[] merged_cut.evt[PI=110:995]
  Name of output events file:[] merged_cut_optimal.evt
```

Or with psrpipe.py:

```
> psrpipe.py --emin <min energy> --emax <max energy> ...
```

Make template

Now make template

```
> nitemplate [options] input_file
```

For example, for events binned into 64 histograms, and to output a profile with 32 phase bins:

```
> nitemplate --nhistbins 64 --nprofbins 32 merged_cut_optimal.evt
```

Note: using unbinned fitting can take a while, so binning is typically recommended

Demo: nitemplate

```
(pint) hesenode1%
(pint) hesenode1%
(pint) hesenode1% ls
0020020101_pipe  0020020109_pipe  1020020101_pipe  6020020104_pipe        J1513-5908_PKS.par
0020020102_pipe  0020020110_pipe  1020020102_pipe  6020020105_pipe        merged_2-10keV_cut.lcurve
0020020103_pipe  0020020111_pipe  1020020103_pipe  cleanfiles.txt         merged_2-10keV.lc
0020020104_pipe  0020020112_pipe  1020020104_pipe  files.txt              merged_cut.evt
0020020105_pipe  0020020113_pipe  1020020105_pipe  files.txt~             merged_cut_optimal.evt
0020020106_pipe  0020020114_pipe  6020020101_pipe  gti_data_squeezed.txt  merged_cut_profinfo.yml
0020020107_pipe  0020020115_pipe  6020020102_pipe  gti_data.txt           merged.evt
0020020108_pipe  0020020116_pipe  6020020103_pipe  gti.fits               OptimalProfile_1.10keV_9.95keV.png
(pint) hesenode1%
(pint) hesenode1%
(pint) hesenode1%
```

Generate TOAs…

Barycentric TOAs:

```
> photon_toa.py --orbfile @orbfiles.txt --ephem DE405 --plot
--plotfile phaseogram.png --tint 500 --outfile toas.tim
merged_cut_optimal.evt itemplate.gauss J1513-5908_PKS.par
```

Topocentric TOAs (needed for long-term effects, with long dataset):

```
> photon_toa.py --orbfile @orbfiles.txt --ephem DE405 --topo
--plot --plotfile phaseogram.png --tint 500 --outfile toas.tim
merged_cut_optimal.evt itemplate.gauss J1513-5908_PKS.par
```

Note: `--tint` value will vary between pulsars, e.g. depending on flux level

Barycentric vs. topocentric timfiles

Topocentric

```
photon_toa 0.000000 57934.9592382679479051 1056.117 STL_GEO  -nsrc 1667.23 -nbkg 7055.77 -expos\
ure 640.0 -dphi 0.00063 -mjdtt 57934.96003901 -telx -6104.97490069 -tely 2857.96780109 -telz -7\
97.07490027 -vx -1.39959292 -vy -4.63629030 -vz -5.93978842 -htest 371.65 -t NICER
photon_toa 0.000000 57935.2825710407212731 773.976 STL_GEO  -nsrc 2676.78 -nbkg 11328.22 -expos\
ure 960.0 -dphi -0.00429 -mjdtt 57935.28337178 -telx -6166.98019422 -tely 2262.51469019 -telz -\
1712.06318512 -vx -0.30463059 -vy -5.12654011 -vz -5.68713461 -htest 620.75 -t NICER
photon_toa 0.000000 57935.4736814040257407 1147.935 STL_GEO  -nsrc 1550.26 -nbkg 6560.74 -expos\
ure 576.0 -dphi 0.02091 -mjdtt 57935.47448214 -telx -5973.00529477 -tely 3121.57885813 -telz -8\
06.19580736 -vx -1.59465397 -vy -4.57552409 -vz -5.93802334 -htest 273.53 -t NICER
photon_toa 0.000000 57936.2470139430570718 793.204 STL_GEO  -nsrc 2649.44 -nbkg 11212.56 -expos\
ure 928.0 -dphi -0.00412 -mjdtt 57936.24781468 -telx -5948.91548078 -tely 2816.46313049 -telz -\
1661.54481272 -vx -0.80123673 -vy -5.05178724 -vz -5.70564436 -htest 536.64 -t NICER
```

Barycentric

```
photon_toa 0.000000 57934.9637210961460247 1056.097 @  -nsrc 1667.23 -nbkg 7055.77 -exposure 64\
0.0 -dphi 0.00063 -htest 371.65 -t NICER
photon_toa 0.000000 57935.2870397931529061 773.960 @  -nsrc 2676.78 -nbkg 11328.22 -exposure 96\
0.0 -dphi -0.00429 -htest 620.75 -t NICER
photon_toa 0.000000 57935.4781417168941433 1147.913 @  -nsrc 1550.26 -nbkg 6560.74 -exposure 57\
6.0 -dphi 0.02091 -htest 273.53 -t NICER
photon_toa 0.000000 57936.2514399364904626 793.187 @  -nsrc 2649.44 -nbkg 11212.56 -exposure 92\
8.0 -dphi -0.00412 -htest 536.64 -t NICER
photon_toa 0.000000 57937.9876579888261973 751.336 @  -nsrc 2510.68 -nbkg 10625.32 -exposure 92\
8.0 -dphi -0.00459 -htest 573.92 -t NICER
```

Note: full observatory list [here]

...and fit with PINT (or Tempo2*)

Command line:

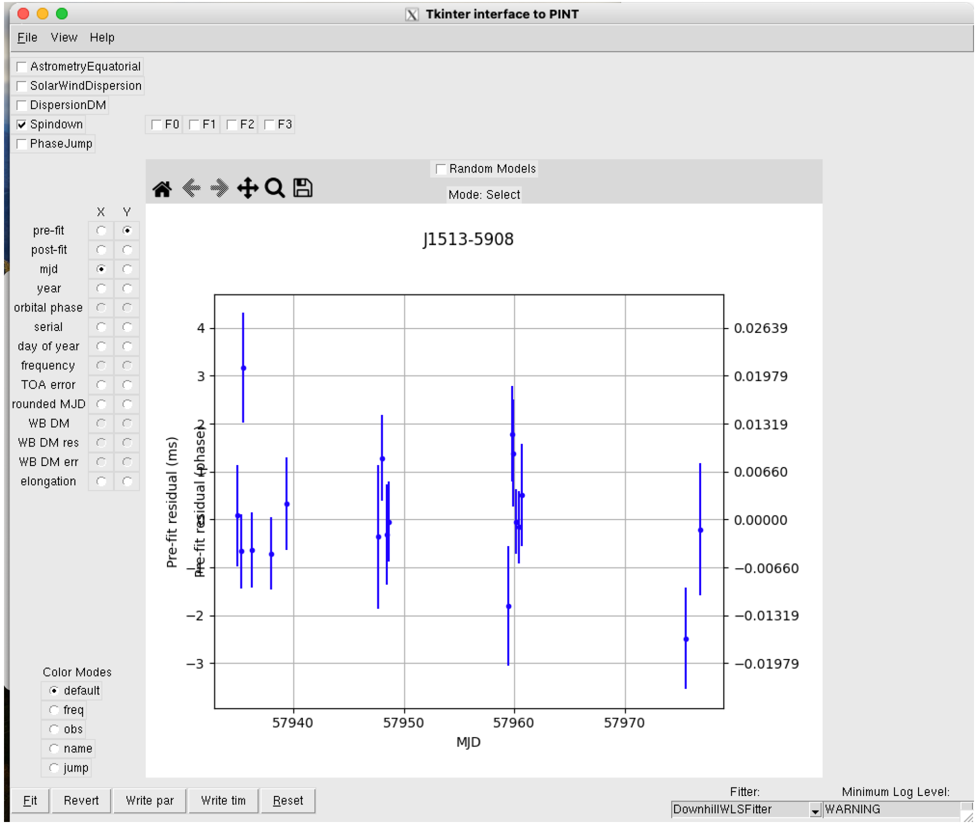
```
> pintk parfile timfile
```

You can also run PINT in python or in a Jupyter notebook.

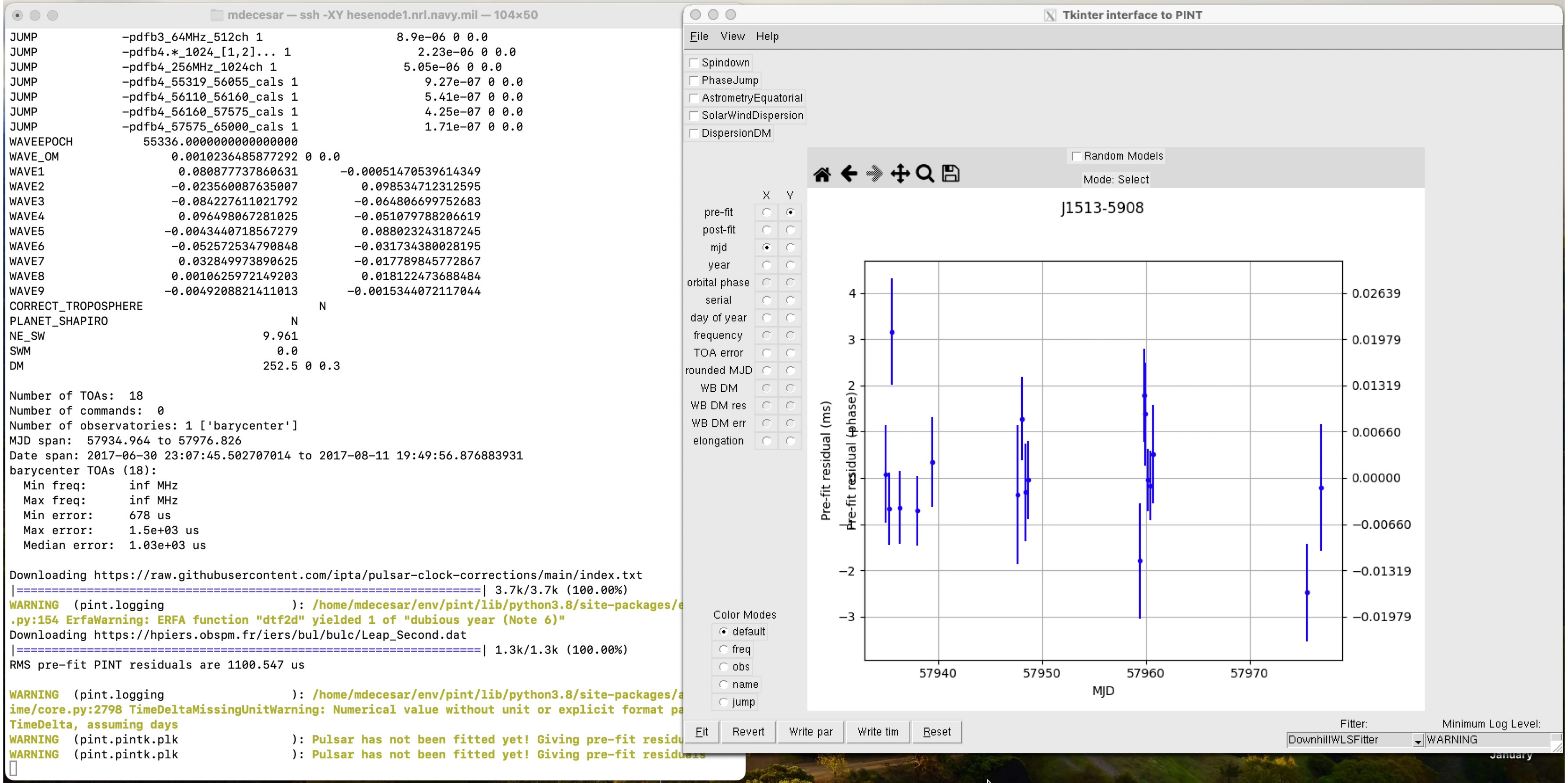
*Tempo2 will also work, but only for barycentric TOAs

Initial parfile:

PSRJ	J1513-5908	
RAJ	15:13:55.6713462	
DECJ	-59:08:09.16532	
F0	6.5970918979437165796	
F1	-6.6530805250943988541e-11	
PEPOCH	55336	
POSEPOCH	54710	
START	54220.401890338307769	1
FINISH	58469.808981595782623	1
TZRMJD	57675.99293714394800	
TZRSITE	pks	
TRES	1054.132	
EPHVER	2	
CLK	TT(TAI)	
UNITS	TDB	



pintk example



Summary

- Pulsar timing yields high precision on pulsar parameters and allows probing of pulsar properties
- There are many tools available for pulsar timing analysis
- This tutorial used NICER data, but the general process is the same no matter what observing frequency, and some tools can be used for different situations (e.g. different space missions, ground-based observatories)
- I hope this tutorial is helpful for those who want to get started on NICER+PINT pulsar timing!

Thanks!

Contact: Megan DeCesar
megan.e.decesar.ctr@us.navy.mil
mdecesa@gmu.edu