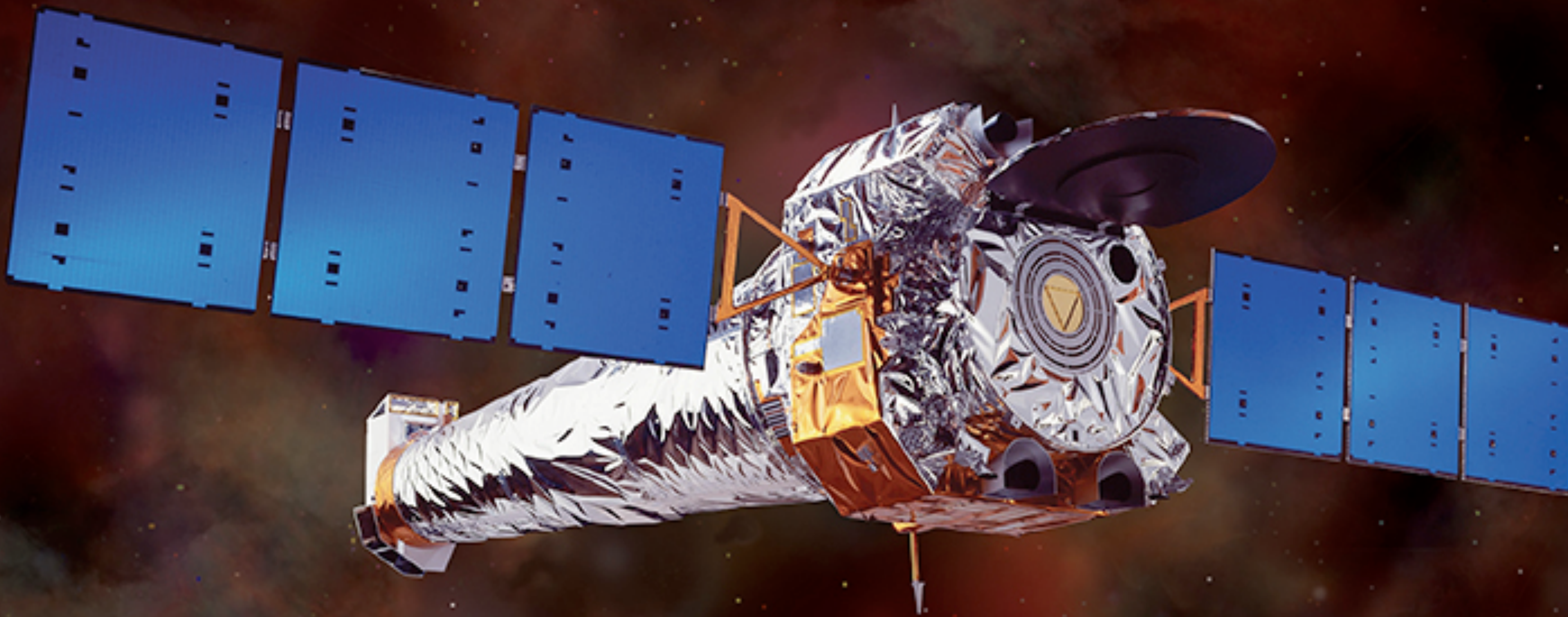
Chandra X-ray Observatory



Timing Analysis

Dr. Michael Nowak
Chandra X-ray Science Center

CHANDRA TIMES: ACIS, TE MODE

- Frame Time (see *Proposer's Guide*):

  T (msec) = (41 + 0.040*q)*m + 2.84*n + 5.2
  q = # of rows from readout
  m = # of active CCDs
  n = # of rows read

- Reality, the Frame Time is an integer multiple of 0.1 sec + frame transfer: (0.2 – 10) sec + 41.04 msec
- Caveat: Images are transferred (quasi-) serially, so there can be up to a 5*41.04 msec delay between CCDs
- Event times are the *middle* of a "frame"

CHANDRA TIMES: ACIS, TE MODE

- Frames take 41.04 msec to transfer to the readout, so there is a certain amount of "dead time" per frame
- Charge is moved at 40 **μ**sec/row, so "readout streaks" potentially perform fast timing of bright sources
- Times are Terrestrial Time, referenced to:

MJD = 50814.0 (0:00 January 1, 1998)
MJD = Julian Date – 2,400,000.**5**

ACIS TIME KEYWORDS

- MJDREF = 50814.
- TIMEZERO = 0. (i.e., corrections to TIME)
- TSTART = start time in seconds from MJDREF
- TSTOP = stop time in seconds from MJDREF
- TIMEPIXR = 0.5 (times are from middle of frame)
- TIMEDEL = Nominal Frame Time
- EXPTIME = Nominal “Live Time”
- DTCOR = EXPTIME/TIMEDEL
- ONTIME_n = per chip quantities
- LIVETIME_n
- EXPOSURE_n

ACIS, TE MODE

- ACIS Clock is stable to 1 part in 10^5, ~1 sec drift over 100 ksec observation
- Times are corrected on the ground to **μ**sec levels (quantized to 10 **μ**sec in event file)
- Time between frames ~ TIMEDEL X (~1 ± 10^{-5})
- Plotting EXPNO vs. TIME usually gives a linear correlation

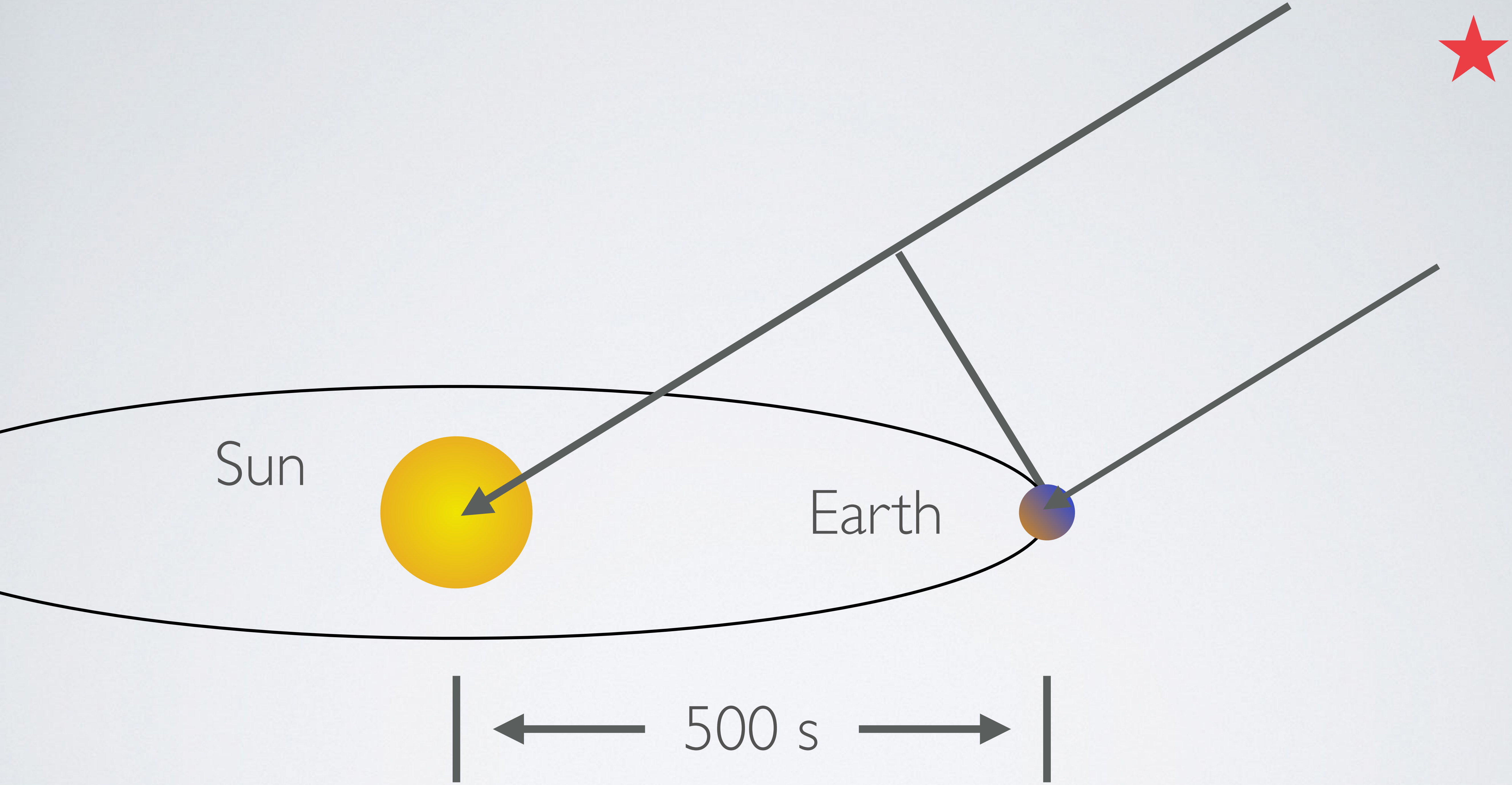
ACIS, CC MODE

- Rows are read out every 2.85 msec
- CCDs are read in parallel
- Time is arrival time, correcting for readout delay from aimpoint, dither, etc.
  - The absolute time could be incorrect, if the position is incorrect. Reprocessing required if position changed.
- 40 **μ**sec rowshift "deadtime" applies

HRC-S

- HRC-I wiring problem limits time accuracy to ~4 msec
- HRC-S can achieve 16 **μ**sec accuracy
- Faster timing than ACIS, but more severe telemetry limits (180 cps), with higher backgrounds
- But, minimal deadtime, and no pileup. Can handle up to 5 cps for a point source.
    - X-ray msec pulsar in a crowded field, use HRC-S!

BARYCENTER CORRECTION



Correct the Time to a Common Location: Barycenter

EXAMPLE: OBS ID 1925

- Quiescent Neutron Star: 4U 2129+47

```
dmcopy \
   "acisf01925N003_evt2.fits[events][(x,y)=circle(4115.1,4167.2,4)]" \
   4U2129_evt2.fits option=all
```

time[0] = 9.201960925809942e+07

- Barycenter correction using axbary

```
axbary infile=4U2129_evt2.fits outfile=4U2129_bary_evt2.fits \
   orbitfile=orbitf091973100N001_eph1.fits
   ra=322.8592 dec=47.2902
```

time[0] = 9.201966493583098e+07

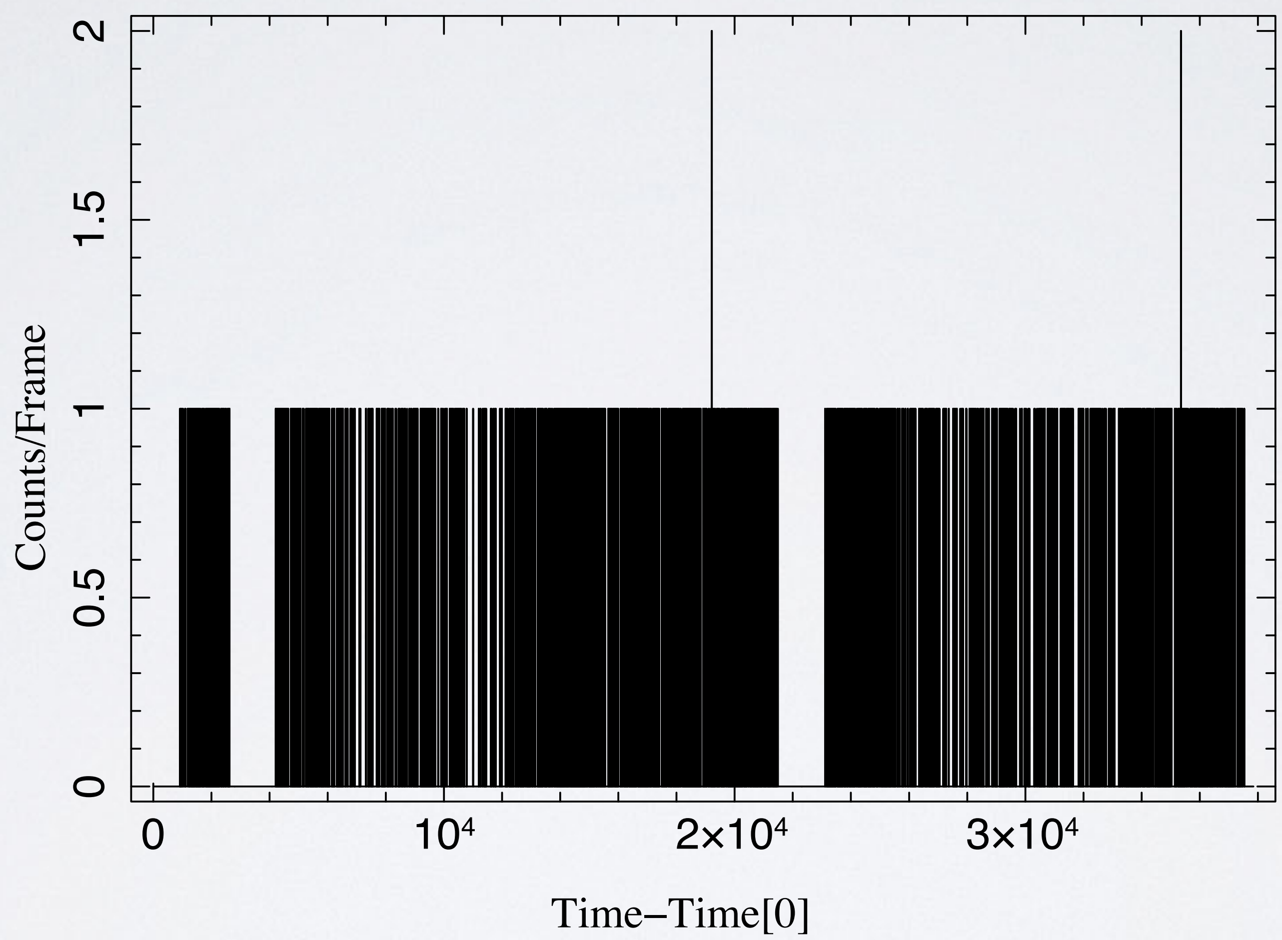
Difference: 55.678 sec

TIMING ANALYSIS QUESTIONS:

- Does my source vary?
- On what time scales does it vary?
- Are the variations periodic or aperiodic?
- How do the variations in different energy bands relate to one another?

CREATING A LIGHTCURVE

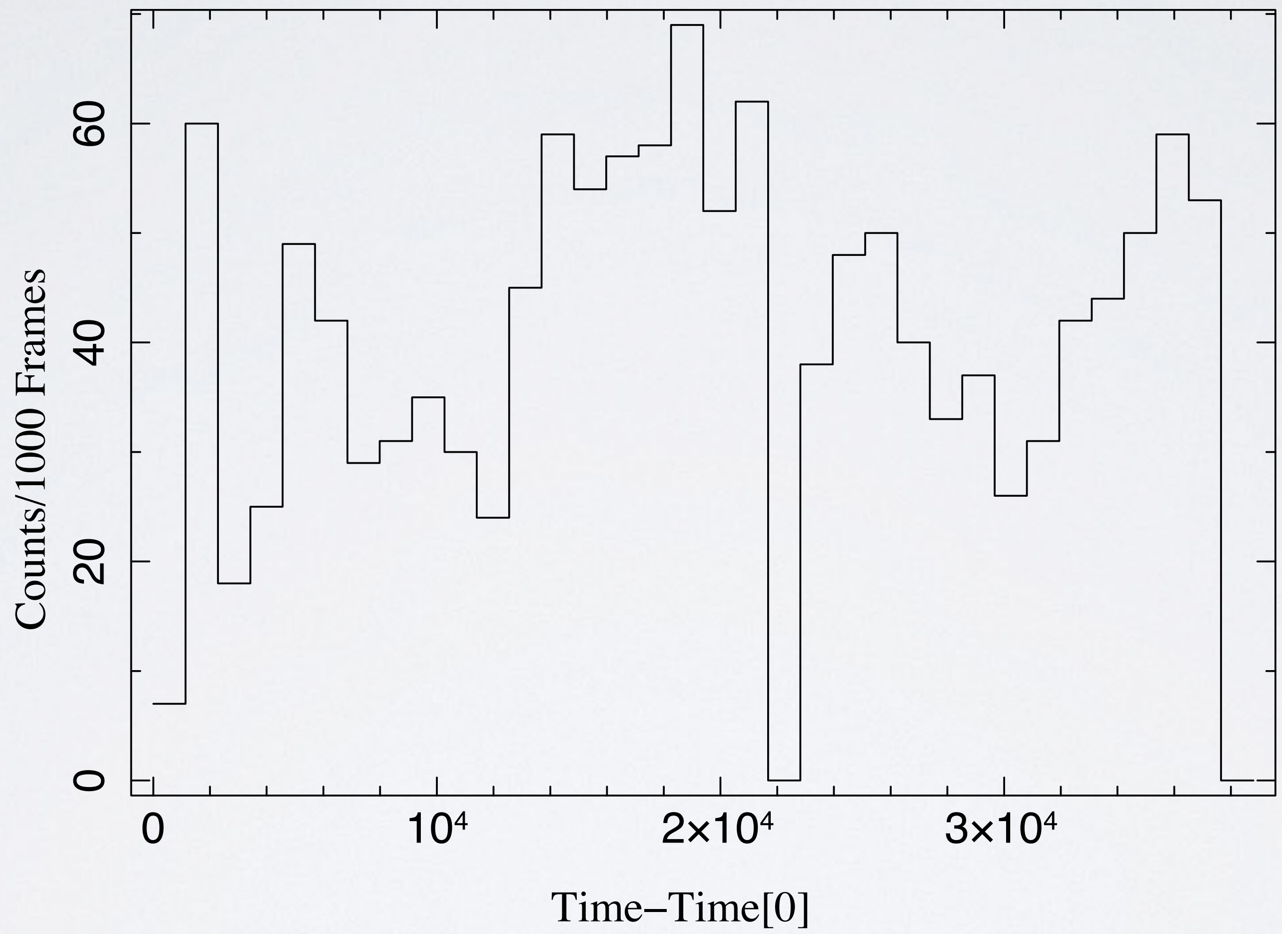
dmextract \
infile="4U2129_bary_evt2.fits[EVENTS][bin time=::1.14096]" \
outfile=4U2129_lc.fits opt=ltc1



- Choose an integer value of a "natural time" unit.
  - Above time is different than TIMEDEL=1.14104 sec (clock drift)
- Consider avoiding binning at all and apply Bayesian tests.

CREATING A LIGHTCURVE

```
dmextract \
  infile="4U2129_bary_evt2.fits[EVENTS][bin time=::1140.96]"   \
  outfile=4U2129_lc_1000.fits opt=ltc1
```

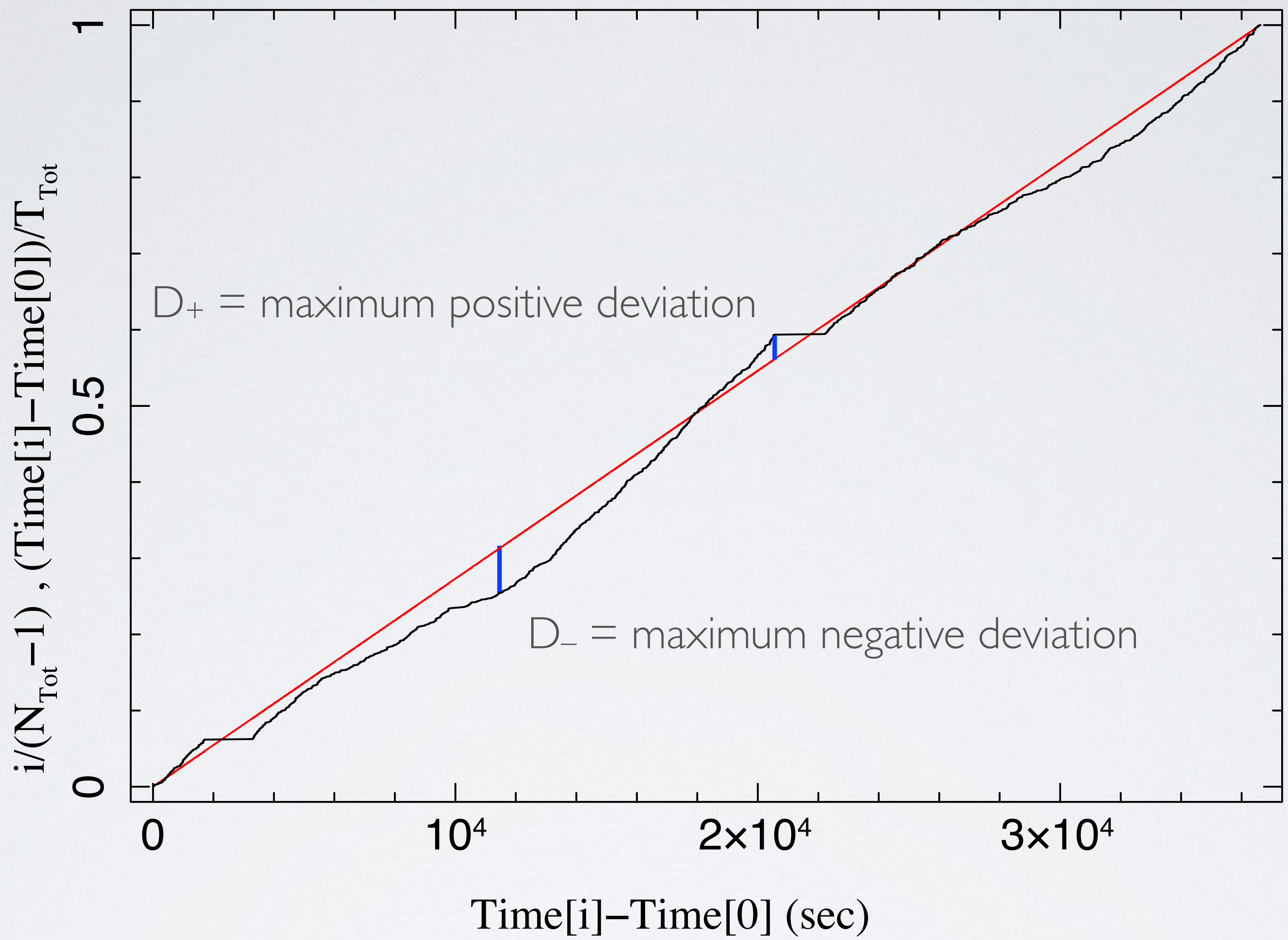


- Remember that dither periods are 707.1 s/1000 s in ACIS, and 768.6/1087 s in HRC, so beware variability on those time scales!

TOOLS OF THE TRADE:

- Unlike spectroscopy, there aren't (yet) standard packages to answer these questions (to everyone's satisfaction)
  - XRONOS: HEASOFT tool, very little used – https://heasarc.gsfc.nasa.gov/xanadu/xronos/xronos.html
  - Stingray: Python package under development – https://stingraysoftware.github.io/
  - SITAR: S-lang/ISIS package I wrote & use – http://space.mit.edu/cxc/analysis/SITAR/
  - Users write their own software
- We provide some useful variability analysis tools in CIAO

KOLMOGOROV-SMIRNOV & KUIPER TESTS



isis> require("stats");

isis> t = fits_read_col("4U2129_bary_evt2.fits","time");

isis> ks_test((t-t[0])/(t[-1]-t[0]));

4.916659859988126e-05

isis> kuiper_test(t-t[0])/(t[-1]-t[0]));

1.5199238953299102e-09

SIMPLE VARIABILITY TESTS

- Kolmogorov-Smirnov, Kuiper tests and variants (e.g., Andersen-Darling; statistically preferred by some) answer "Yes/No" Question:
  - "Is this consistent with a constant?"
  - They do not *characterize* variability
- Many sources for Python, R, IDL, … code for such tests
- You may have to include variation in your extracted region, e.g., dithering past chip edge, using the **dither_region** tool
- We provide one CIAO tool, **glvary,** that does *characterize* variability.

GLVARY

- See thread: http://cxc.harvard.edu/ciao/threads/variable/
- Bayesian technique based upon Gregory & Loredo 1996, ApJ, 473, 1059

```
unix%> punlearn ardlib
unix%> acis_set_ardlib acisf01925_002N003_bpix1.fits
unix%> punlearn dither_region
unix%> pset dither_region region="region(ds9.reg)")
unix%> pset dither_region infile=pcadf092019600N003_asol1.fits
unix%> pset dither_region maskfile=../secondary/acisf01925_002N003_msk1.fits
unix%> pset dither_region outfile=fracarea.fits
unix%> dither_region
unix%> dmkeypar 4U2129_bary_evt2.fits DTCOR echo+
0.96403281217135
unix%> dmtcalc "fracarea.fits[cols time,fracarea]" dtf_fracarea.fits \
expression="dtf=(0.96403281217135*fracarea)" clob+
```

GLVARY

- See thread: http://cxc.harvard.edu/ciao/threads/variable/
- Bayesian technique based upon Gregory & Loredo 1996, ApJ, 473, 1059

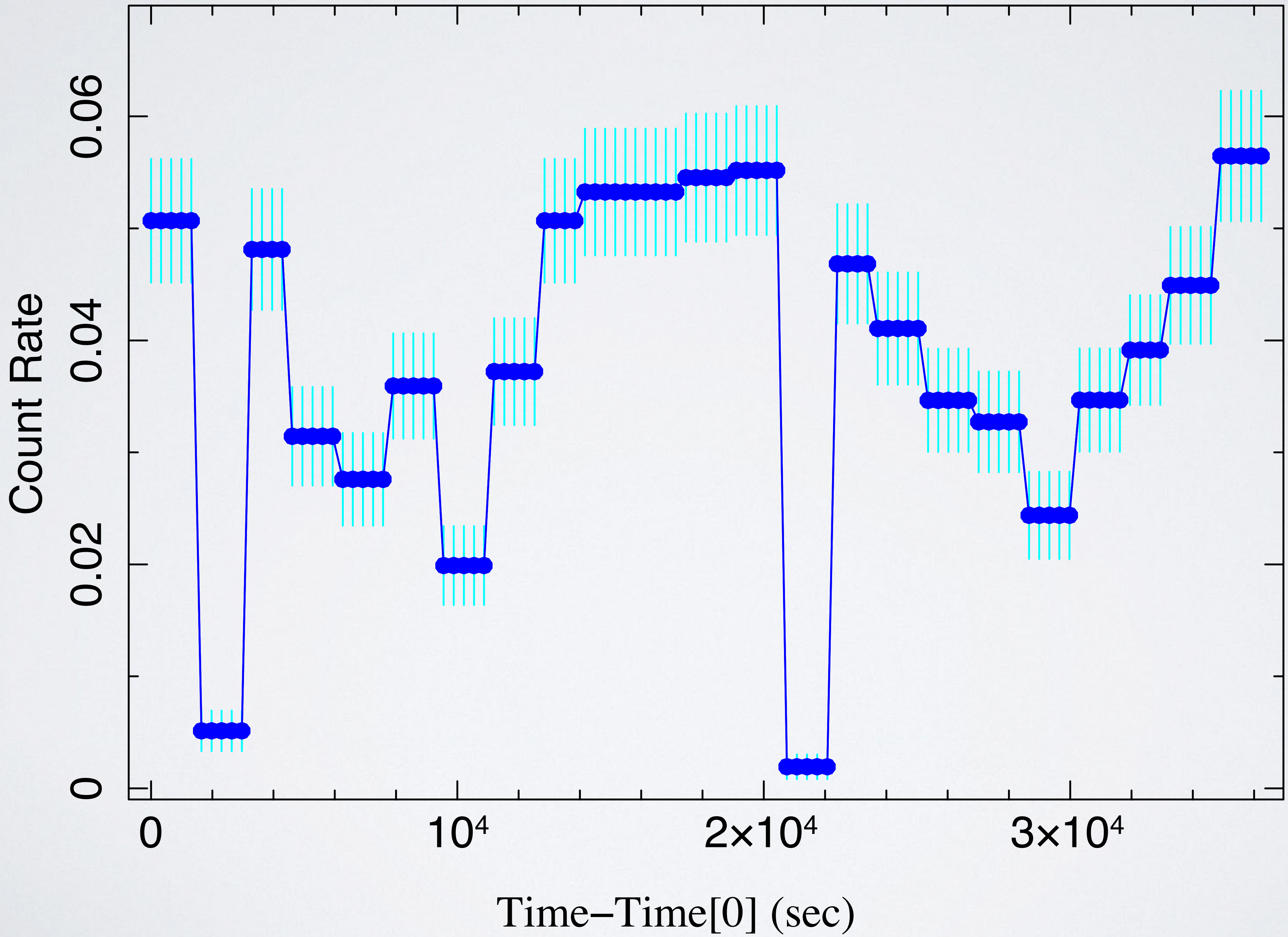
```
unix%> punlearn glvary
unix%> pset glvary infile="4U2129_bary_evt2.fits[sky=region(ds9.reg),ccd_id=7]"
unix%> pset glvary effile=dtf_fracarea.fits
unix%> pset glvary outfile=gl_prob.fits
unix%> pset glvary lcfile=lc_prob.fits
unix%> glvary

unix%> dmlist gl_prob.fits header | grep -i variab
```

0008 ODDS	35.8132362773	Real8	Odds for variable signal 10Log
0009 PROB	1.0	Real8	Probability of variable signal
0014 VARINDEX	10	Int4	Variability index

- Plot TIME, COUNT_RATE, COUNT_RATE_ERR columns from lc_prob.fits file

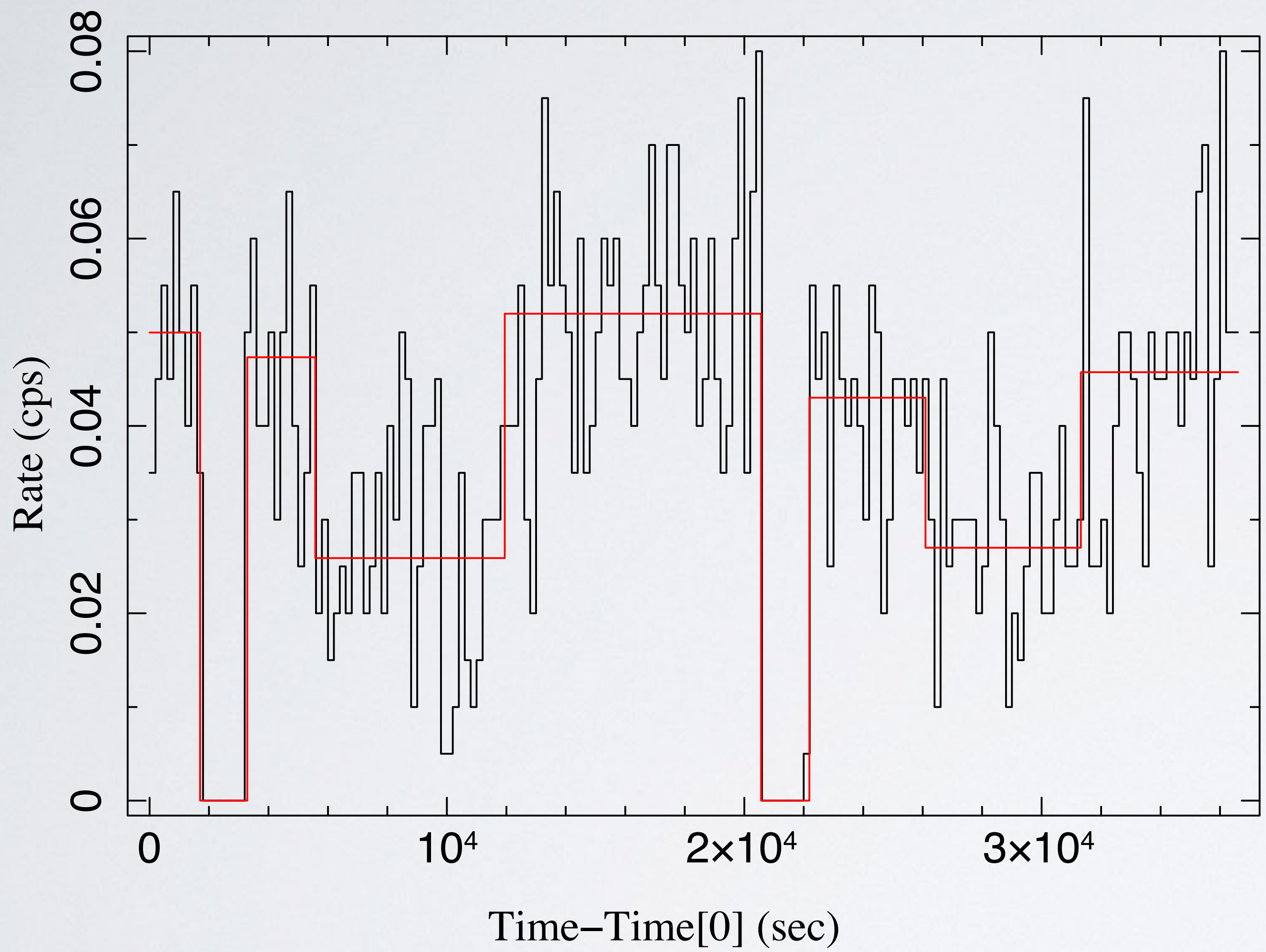
GLVARY

- **glvary** forces a *weighted combination* of evenly binned lightcurves, and *doesn't* vary the phasing of these bins.



BAYESIAN BLOCKS

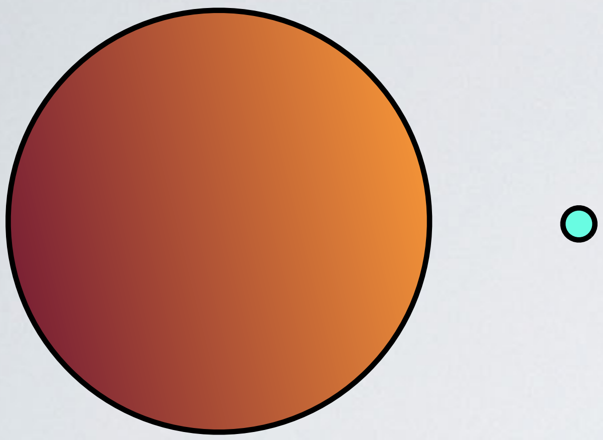
- Better captures eclipse. S-lang, Python, etc., code can be found.




```
isis> () = evalfile("sitar.sl");
isis> t =
fits_read_col( "4U2129_bary_evt2.fits",
            "TIME" );
isis> cell = sitar_make_data_cells( t-t[0], 3, 0.5,
              1.14096, 0., t[-1]-t[0] );
isis> bb = sitar_global_optimum( cell, 3., 3 );
isis> lc = sitar_bin_events( t-t[0], 200., [0.],
             [t[-1]-t[0] );
isis> hplot( lc.bin_lo, lc.bin_hi, lc.rate );
isis> ohplot( bb.lo_t, bb.hi_t, bb.rate );
```

Each block >99.8% significant.

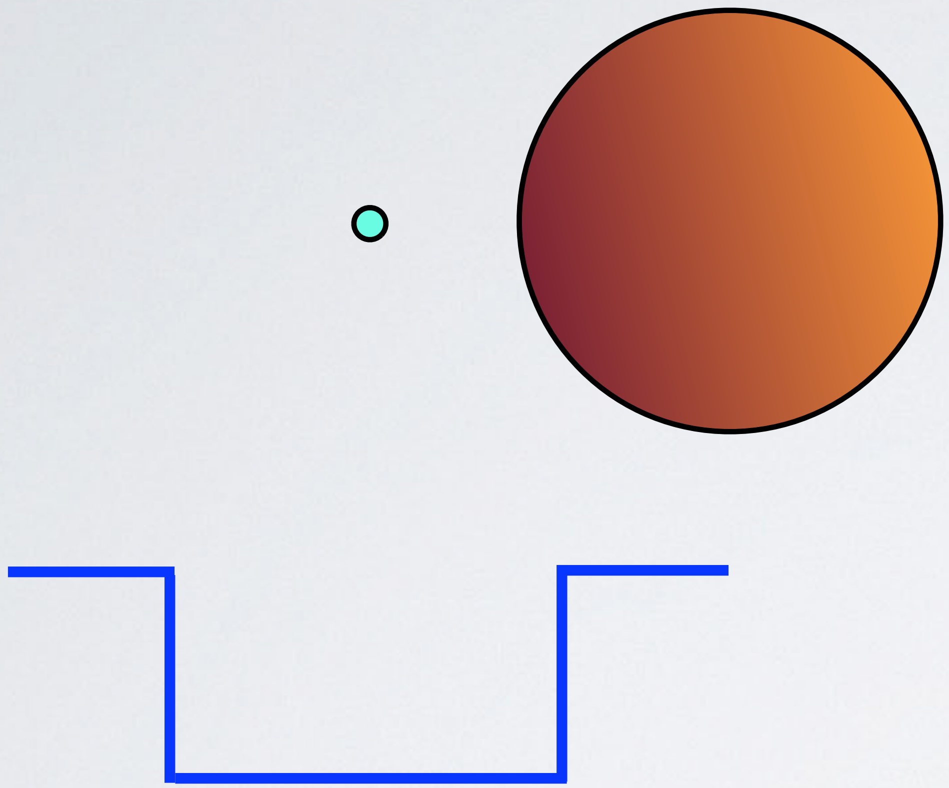
- Blocks can be uneven, so it's a good for capturing flares (Sgr A*). Method developed by Scargle et al. 2013, ApJ, 764, 167.



- Eclipses are consistent with & modeled as instantaneous
- Chandra has virtually no background
- Last/First event seen yields eclipse duration & midpoint

$$P(\Delta t_{cross}) = R\ exp(-R\Delta t_{cross})$$

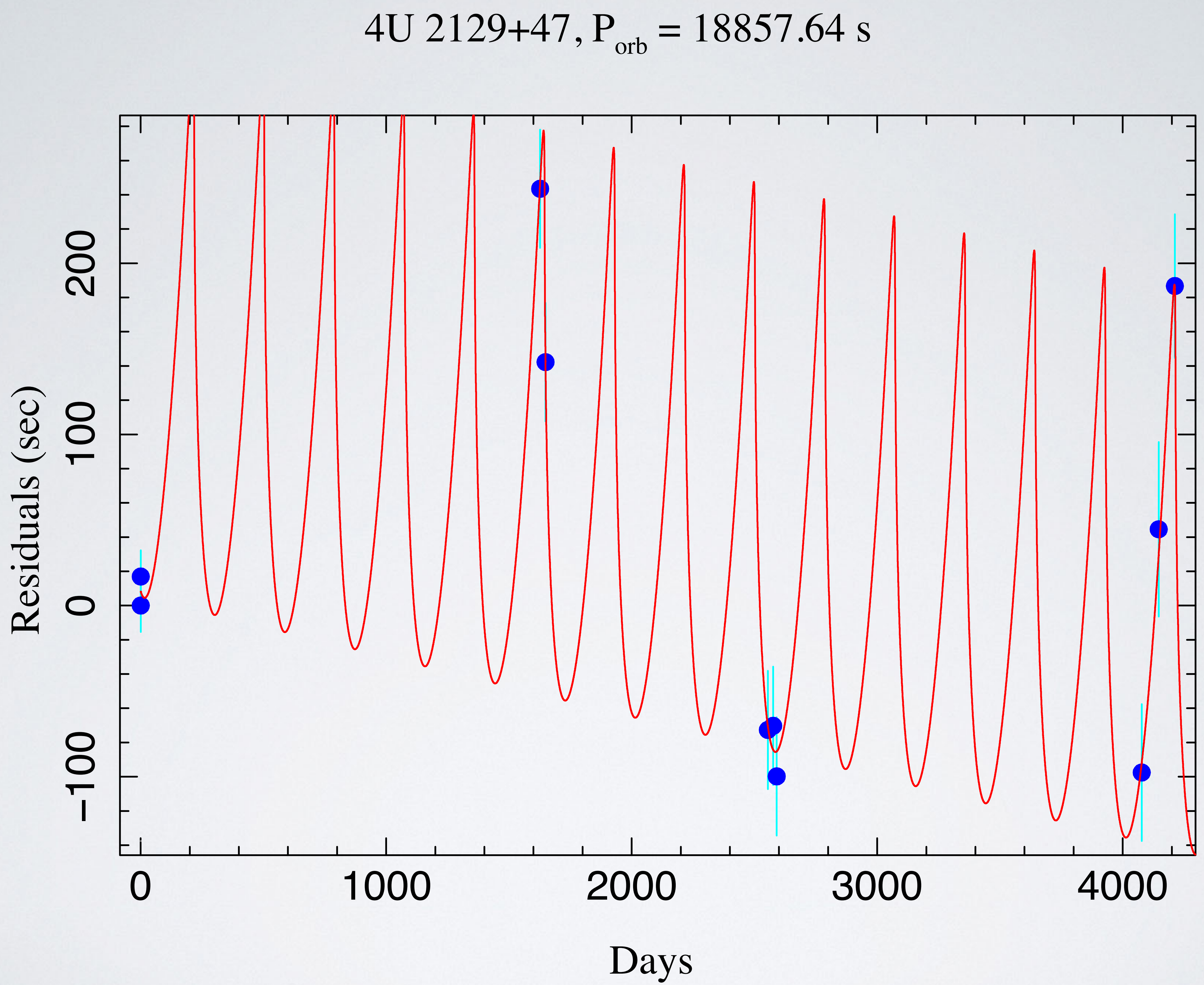
- Can be modified for expectation for rate on either side of the eclipse, deadtime, & background (Nowak, Heinz, & Begelman 2002, ApJ, 573, 788)
- *No binning was done!*



- Eclipses are consistent with & modeled as instantaneous
- Chandra has virtually no background
- Last/First event seen yields eclipse duration & midpoint

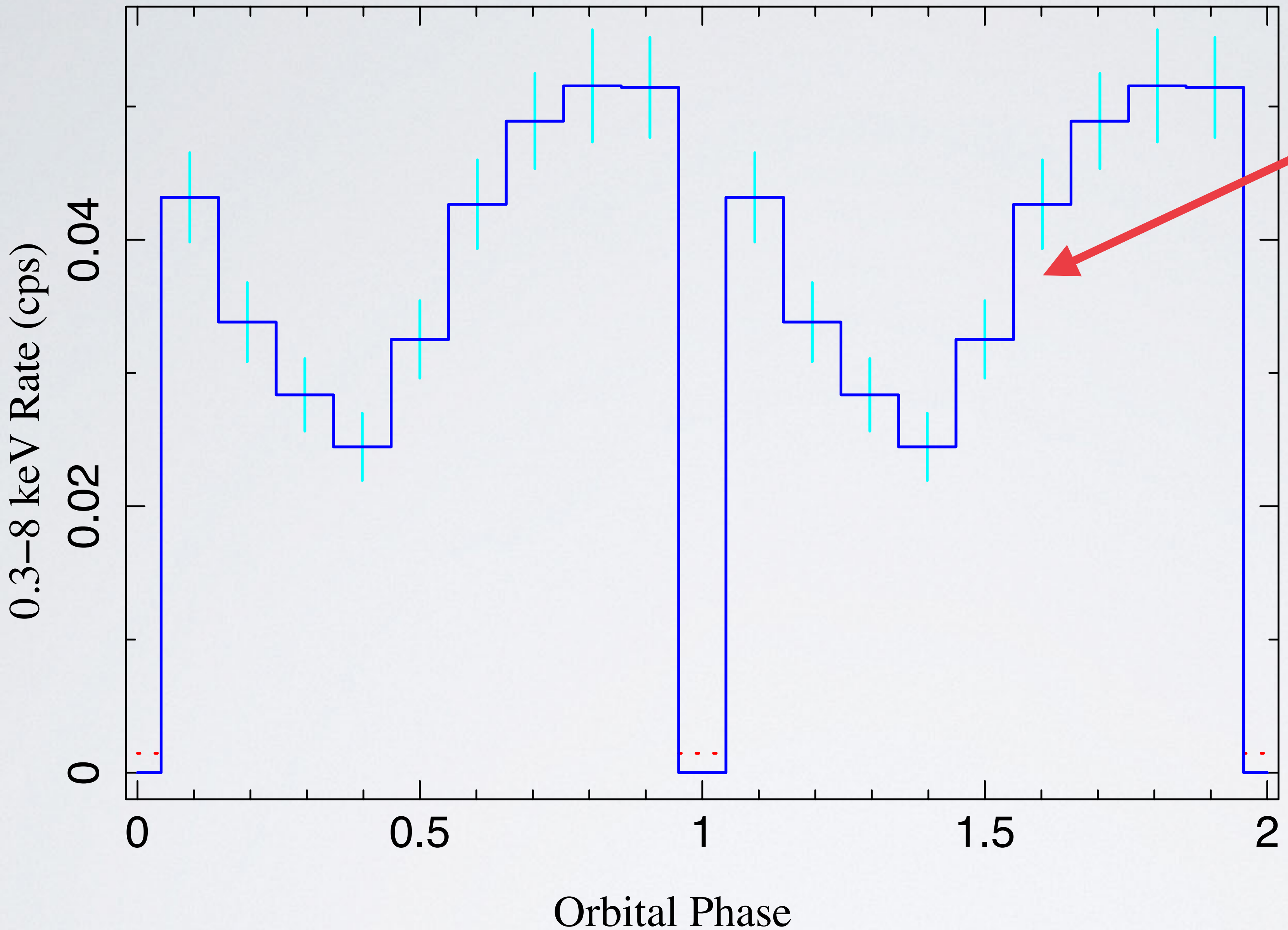
$$P(\Delta t_{cross}) = R\ exp(-R\Delta t_{cross})$$

- Can be modified for expectation for rate on either side of the eclipse, deadtime, & background (Nowak, Heinz, & Begelman 2002, ApJ, 573, 788)
- *No binning was done!*



Multiple Chandra (& XMM) ⇒ 3rd Body

YOU CAN FIT YOUR LIGHTCURVE



ISIS used to fit sinusoid with (35%±6%) amplitude (90% CL)

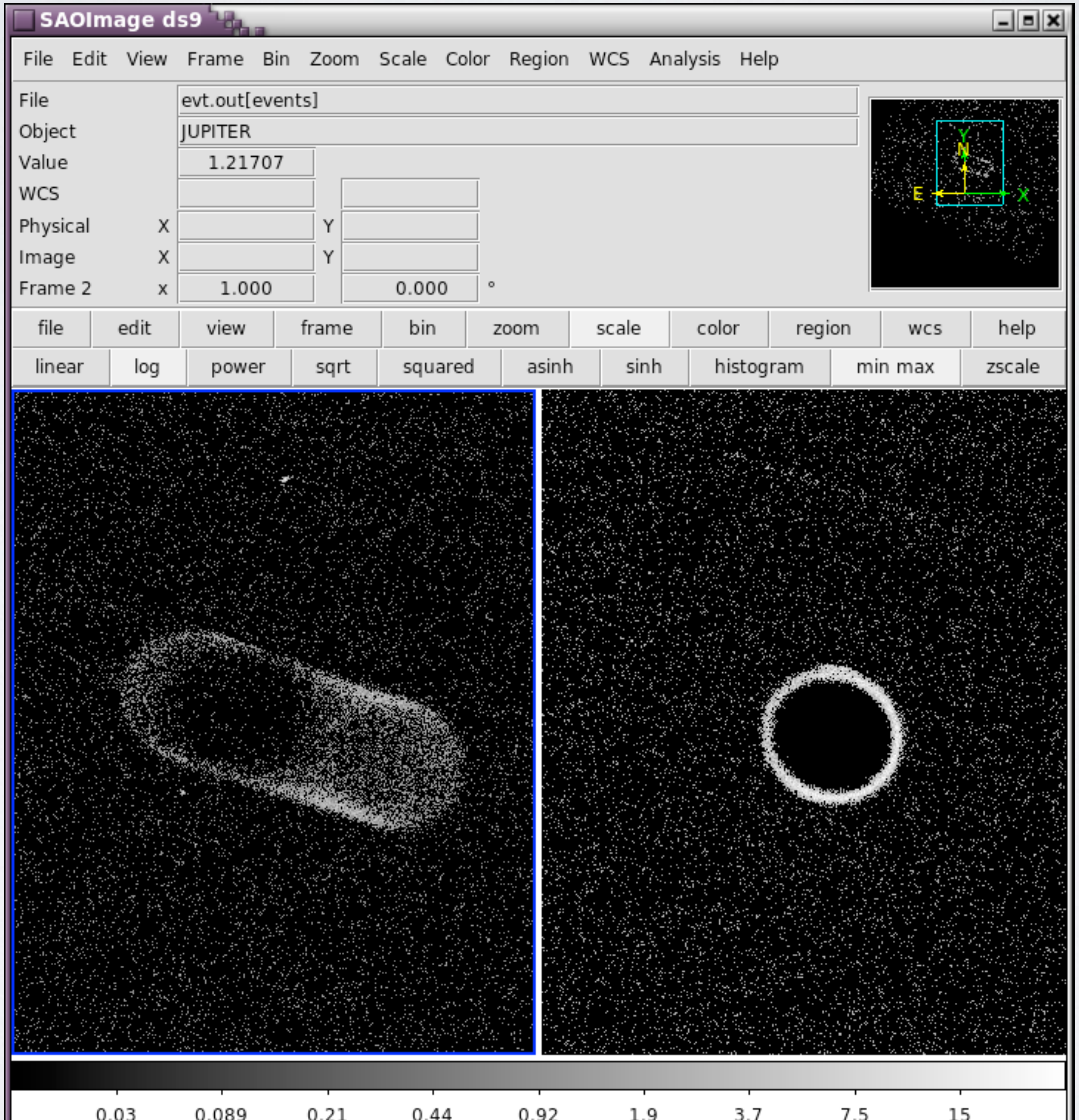
- Very easy in **ISIS** or **Sherpa** to make a counts/bin lightcurve a fittable dataset & create simple models to describe.
  - Somewhat more complex in **XSPEC**, but possible

LONG TIME SCALE: CRAB

LONG TIME SCALE: CRAB

INTRA-OBSERVATION MOTION: SOLAR SYSTEM

ACIS-S View of Jupiter



Before Correction

After Correction

SSO_FREEZE

- See thread: http://cxc.cfa.harvard.edu/ciao/threads/ssofreeze/

```
unix% download_chandra_obsid 1463 evt2,asol,eph1
unix% ls *eph1.fits
jupiterf059875200N001_eph1.fits
jupiterf059875200N002_eph1.fits
orbitf059443264N002_eph1.fits
unix% dmkeypar acisf01463N006_evt2.fits TSTART echo+
59968797.485273
unix% cat pcad_asol1.lis
pcadf059968984N004_asol1.fits

unix% punlearn sso_freeze
unix% pset sso_freeze infile=acisf01463N006_evt2.fits
unix% pset sso_freeze scephemfile=orbitf059443264N002_eph1.fits
unix% pset sso_freeze ssoephemfile=jupiterf059875200N002_eph1.fits
unix% pset sso_freeze asolfile=@pcad_asol1.lis
unix% pset sso_freeze ocsolfile=1463_oc_asol1.fits
unix% pset sso_freeze outfile=frozenjupiter.fits
unix% sso_freeze
```

GRATINGS LIGHTCURVE COMPLICATIONS

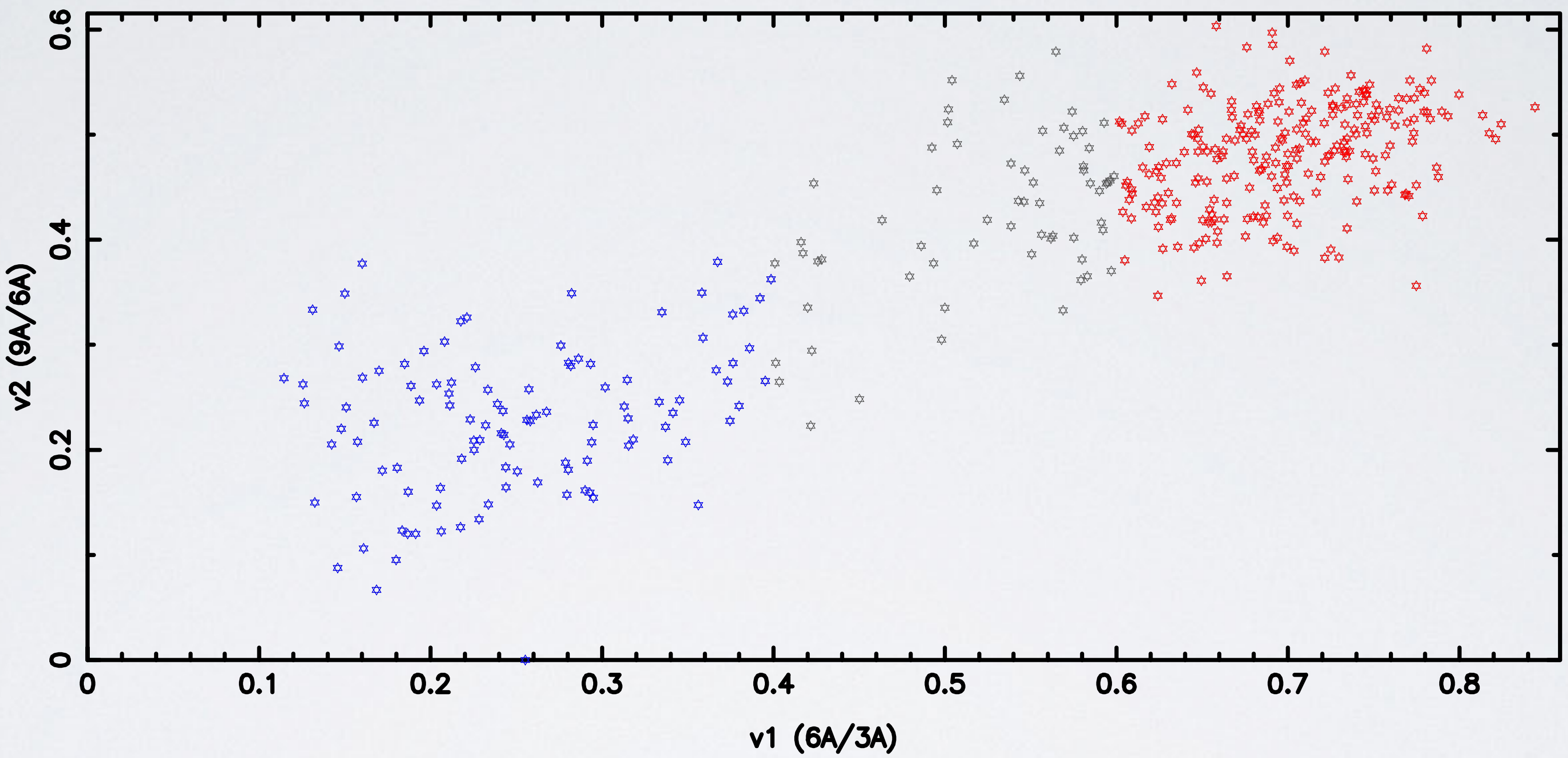
- Spatially extended –
  - Regions dither across chip edges and "nodes"
  - Not intuitive how to translate from wavelength to spatial extraction window
  - Same wavelength regions on the four arms (MEG/HEG, $\pm 1^{st}$ orders) are different spatial regions
- For Timed Exposure (TE) mode –
  - Slight offsets in the times from CCD to CCD

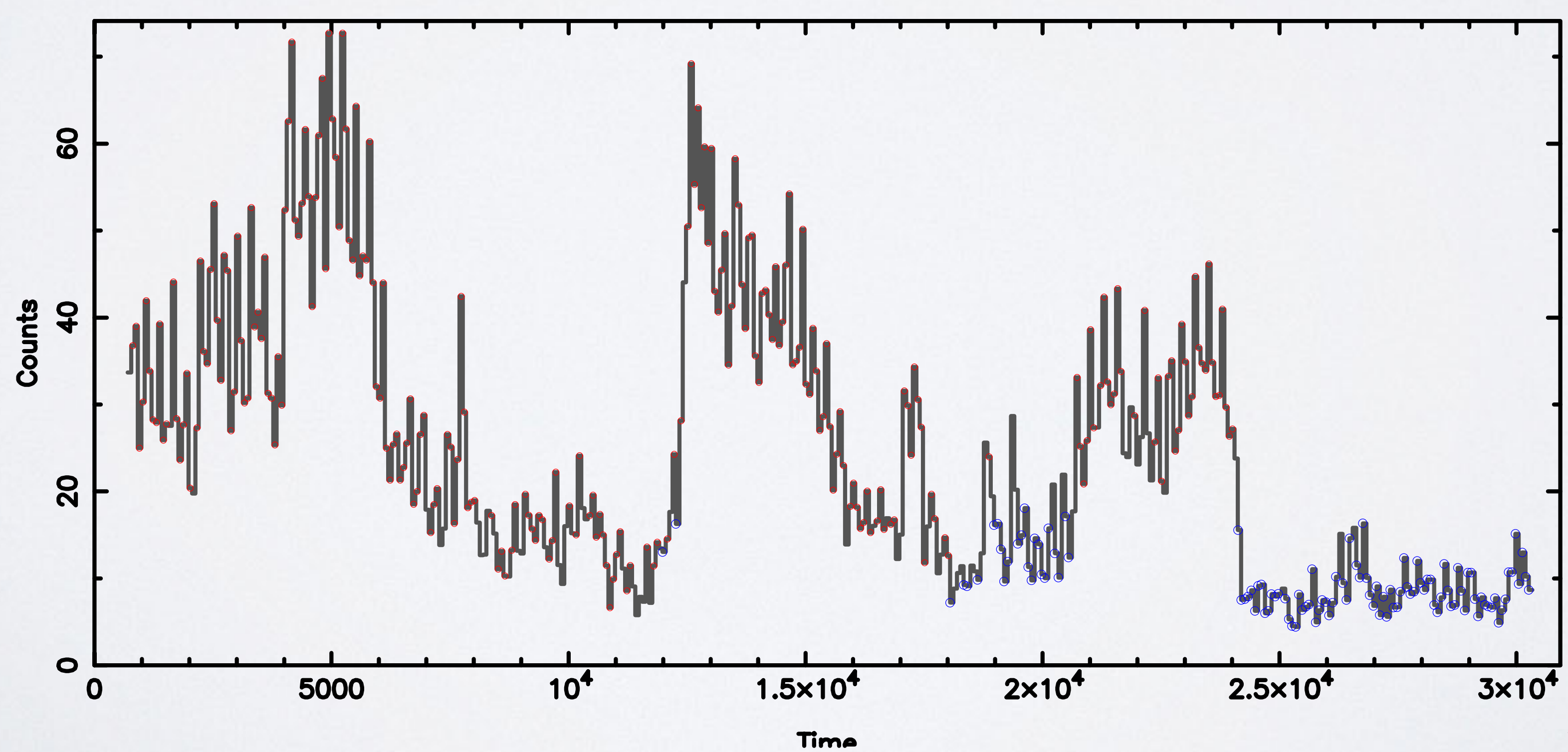
AGLC
(ACIS GRATING LIGHT CURVE)

- S-lang/ISIS suite of tools by Dave Huenemoerder (MIT) – http://space.mit.edu/cxc/analysis/aglc/aglc.html
- Fundamentally uses EXPNO as an implicit time coordinate
  - Barycenter correction has to be applied to lightcurve *after* its creation via **aglc** (apply analytic or empirical mapping from uncorrected/barycentered event times)
- Allows one to create rate or period phase-folded lightcurves with properly calculated exposures from multiple CCDs
- E.g., hardness ratios are tricky with gratings, **aglc** can handle them

AGLC EXAMPLE: VELA X-1

- S-lang/ISIS suite of tools by Dave Huenemoerder (MIT) – http://space.mit.edu/cxc/analysis/aglc/aglc.html





AGLC EXAMPLE: VELA X-1

- S-lang/ISIS suite of tools by Dave Huenemoerder (MIT) – http://space.mit.edu/cxc/analysis/aglc/aglc.html