Using fasebin *to Create Phase-Binned Spectra and Histograms*

Recipes from the RXTE Cook Book

- Introduction
- Data files you will need
- Running fasebin
- Combining fasebin files from different Orbit days
- Analyzing fasebin spectra in xspec
- Subtracting persistant emission
- Combining several phase bins into one
- Creating a counts-per-phase-bin histogram

Introduction

The recommended tool for producing phase-binned spectra is **fasebin**. Fasebin handles the extraction, barycenter correction, and phase-binning of input RXTE data, eliminating the need to run separate tools for these various functions. It produces a single FITS file in the OGIP Type 2 spectral format: each row in the output file is a separate spectrum associated with a given phase bin. Other tools in the fasebin suite can then be used to subtract persistent emission (**fbsub**), combine phase-bins (**fbfsum**), and produce a "counts-per-phase-bin" histogram (**fbssum**). Examples of each will be shown below.

Data files you will need

To run **fasebin** you will need to have the following:

- **ascii list(s) of the PCA or HEXTE data files you wish to phase-bin** (note that for HEXTE data, one inputs the original event file; fasebin will take care of extracting only the on-source data from the file). The file list should contain one file name per line, followed by a return with no spaces or blank lines. (File lists produced by **XDF** are in the correct format.) For a list of supported data modes, see "**fhelp fasebin**"; note that all RXTE data modes are not compatible with fasebin at this time. The spectral binning of your output file will be determined by the spectral binning of the input data you choose (eg., GoodXenon data has 256 channels, and will produce output spectra at 256 channel resolution). **DO NOT MIX DATA FROM DIFFERENT CONFIGURATIONS.**

- **FPorbit_DayNNNN** files (Orbit Ephemeris files) corresponding to the PCA or HEXTE files in your data file list. Orbit files are available under the "OrbitEphem" directory on your data tape, or on the Web at:

  `ftp://legacy.gsfc.nasa.gov/xte/data/archive/OrbitEphem`

- **GTI file (optional)** giving good times for this series of data. Only one GTI FITS file may be input per FPorbit day; to merge several GTI files into one use the FTOOL **mgtime**

- **orbital period and binary orbital period** (if applicable) for your source. NOTE: if your source is listed in the Princeton Radio Pulsar Database, an appropriate period for the epoch of your observation will automatically be provided. See "**fhelp fasebin**" for further details ("SYSTEM FILES" section).

Running fasebin

1. **For users analyzing data in GOOD XENON MODE ONLY:**

   Run **make_se** on your data files, to produce full binary FITS files from the Good_Xenon1 and Good_Xenon2 raw data files. Make an ascii list of these new files, and treat this as your list of PCA data files in all future steps.

2. **Determine which PCA or HEXTE data files correspond to which Orbit Ephemeris files** (ie., which observations occurred on which days). Segregate these files into separate ascii lists by orbit day. The end result should be a separate file list for each orbit day, with each list giving only the data files which occurred on that day. (**XDF** can be a useful tool for matching observations to Orbit files; see the RXTE *Getting Started Guide* for details on using XDF.)

3. **Look at the text file:**

   `$LHEA_DATA/psrtime.dat`

   (This directory and file are created automatically when FTOOLS is installed on your system.)

   This file contains a recent copy of the Princeton Radio Pulsar database. If your source is in this file with the correct epoch and period, you may skip steps #4 and 5, and go directly to step 6. (Obviously, if your source is not a radio pulsar it will not be in this file; proceed directly to step **#4**.)

4. **Create a new directory (eg., "fasebin_files") and type the following UNIX command:**

   setenv TIMING_DIR `/full/path/name/fasebin_files`

   where "`/full/path/name/fasebin_files`" is the full UNIX path name of the directory you just created.

5. **Copy the psrtime.dat file into your new directory and edit it to include an entry for the source you wish to analyze.** Eg.:

   cp **`$LHEA_DATA/psrtime.dat $TIMING_DIR/psrtime.dat`**

edit $TIMING_DIR/psrtime.dat

- source entries are in alphabetical order by RA-Dec name (first column, labled "`PSR B`")
- create a new entry for your source by making a source name from its RA and Dec in hexagesimal coordinates; place the entry in the proper location to preserve the ascending order of source names.
- fill in the RA and Dec positions of your source (in J2000)
- place the period in inverse seconds in the "`f0(s^-1)`" column
- the "`MJD1`" and "`MJD2`" columns contain the time span in MJD over which this period is valid. Fill in appropriate values, but **make sure this time span includes the dates of the RXTE observations you wish to analyze**.
- the Web tool **xTime** can be used to convert times into MJD (xTime is also found under the "Proposal and Planning Tools" button on the GOF homepage).
- place the reference time for phase=0 in the "`MJD(t0)`" column. Note this time must be in MJD, and it need *not* fall within the time span specified in "`MJD1`","`MJD2`".
- (optional) if you have a first or second derivative for the period, or an RMS error, fill these in under the appropriate columns
- place the source name again in the final column ("`PSR J`")
- save and exit file

6. **If your source has a binary companion** and you would like to correct for its orbit, open the following file with any text editor:

   `$LHEA_DATA/psrbin.dat`

   This is the binary orbits table for the Princeton Radio Pulsar Database. If your source is listed (under "`PSR B`" column), proceed to **step 7**. If not, create a `$TIMING_DIR` as in step #4 (unless you have already done so), and follow **steps #4 and #5** for file "psrbin.dat". Place the appropriate binary orbit information in your copy of the psrbin.dat file instead of the pulsar period information listed in step #5. (See "**fhelp fasebin**" for more information on the data columns in the Princeton Radio Pulsar Database binary tables.)

7. You're finally ready to run **fasebin**:

   given:

```
orbit file:                    FPorbit_Day1107
list of PCA data files:        binned_data_day1107.txt
GTI file spanning these data:  clean_data.gti
```

type:

```
fasebin

Orbit File[] FPorbit_day1107
Name of output .pha file[faseBin.pha] day1107.pha
Name of GTI file[none] clean_data.gti
Source Name[] 0833-45
Number of phase bins[100] 20
Datafile(s)[] @binned_data_day1107.txt
```

This will produce an output file called "`day1107.pha`" with spectra in 20 phase bins for source "`0833-45`", whose period and other characteristics were read from the "`psrtime.dat`" and "`psrbin.dat`" files.

N.B.: fasebin produces spectra for *2 full phase cycles*. Thus, if you requested **N** phase bins, your output file will contain **2N+1** spectra covering two full phase cycles, with the second cycle being identical to the first.

Good Xenon data only: fasebin automatically extracts data from *only the first (top) layer of the PCA* when Good Xenon data are used. For faint sources (~ < 20 cnt/sec/PCU) which don't have unusually hard spectra, this greatly increases the signal-to-noise of the accumulated spectra. If you would rather collect data from the full PCA (all 3 layers) use this hidden parameter on the command line when you invoke fasebin:

```
fasebin l1only=no
```

See "**fhelp fasebin**" for further details.

Combining fasebin files from different Orbit days

To combine data of the same source and configuration taken over multiple Orbit days:

- repeat step #7 for each Orbit day, until all data from all days has been phase-binned
- create an ascii file list with the names of the all the output phase-binned spectral files, eg:

```
ls day*.pha > alldays_pha.list
more alldays_pha.list
day1107.pha
day1108.pha
day1112.pha
```

- run **fbadd** to make a single phase-binned spectral file combining data from all these days:

```
fbadd
Input File (or @filelist)[] @alldays_pha.list
Name of output .pha file[fBadd.pha] alldays.pha
```

Analyzing fasebin spectra in xspec

Phase-binned spectral files from any of the fasebin tools may be read directly into **xspec 10.0** or higher. (Earlier versions of XSPEC do not support OGIP Type 2 Spectral FITS files.) The syntax for reading Type 2 spectral files in xspec is:

```
XSPEC>  data alldays.pha{10}
```

This will read the 10th spectrum (ie., the tenth row) of your phase-binned .pha file; to display more than one spectrum at a time, use the "`data 1`", "`data 2`" syntax in xspec (see the xspec manual for details).

To produce a response matrix, ***DO NOT*** give your phase-binned spectral file to **pcarsp**. Instead, extract a normal, un-phase-binned spectrum for the same type of data you gave fasebin (eg., if you ran **fasebin** on PCA binned mode data, run **saextrct** on some part of the same binned data and produce a spectrum). Give this un-phase-binned spectrum to **pcarsp** and make a response matrix from this. Since you are using the same data that you gave to fasebin, the response matrix will be valid for the fasebin spectral file as well. (**Good Xenon users only**: keep in mind that fasebin, by default, extracts only layer 1. If you used this default behavior, make sure you specify layer 1 only (`LR1`) when you run pcarsp.)

Subtracting persistant emission

You may subtract the persistent emission (the baseline background) from your data in one easy step, with the FTOOL **fbsub**:

```
fbsub
Input File[] alldays.pha
Name of output .pha file[fBsub.pha] alldays_pulsed-only.pha
Phase minimum[0.0]
Phase maximum[1.0]
```

The phase minimum and phase maximum may contain any continuous range of phase space which you feel best represents the unpulsed component for your source. The entire phase range (0.0 to 1.0) may be used in most cases as a good first approximation, since the few phase bins containing the pulsed emission will not effect the baseline much. You may also enter a second phase range to be used in calculating the unpulsed background -- see "**fhelp fbsub**" for details.

Combining several phase bins into one

You may combine contiguous phase bins into a single spectrum using the FTOOL **fbsum**:

```
fbfsum binstart=5 binstop=8
Input File[] alldays.pha
Name of output .pha file[fBfsum.pha] alldays_phase5-8.pha
Number of phase bins (rows) to be averaged[5] 4
```

This will create a Type 2 spectral file with a single row representing the combined spectrum for phase

bins 5-8. If the last input (number of phase bins) was changed from "4" to "2", the output file would contain two spectra, one for phase bins 5+6, the other for bins 7+8. Leaving out the "`binstart=5 binstop=8`" parameters on the command line would combine every 4 phase bins in the input file into a single spectrum in the output, resulting in an output file with 5 separate spectra. (The input file contains 20 phase bins.)

Creating a counts-per-phase-bin histogram

To produce a histogram of counts-per-phase-bin, use the FTOOL **fbssum**:

```
fbssum
Input File[] alldays.pha
Channels for energy ranges[]
```

This will sum the counts in all channels for each phase bin, and dump the resulting totals to the screen. Header-style comments at the top of the output define the columns and units, as well as echoing the included channels. You may cut and paste the entire output to a file and plot with any ascii plotting package (eg., QDP, gnuplot, etc.).

Please Note: the resultant histogram covers two full phase cycles, ie., 0.0 to 2.0 in phase space. Thus, if you requested N phased bins when running fasebin, you will have 2N+1 phase bins in your output.

For the syntax on creating histograms for restricted channel ranges, see "**fhelp fbssum**". **BEWARE:** ***the channels are relative to the configuration used, NOT THE ABSOLUTE 0 to 256 CHANNEL RANGE OF THE PCA INSTRUMENT***. See the fhelp for further details.