spi_psd_fitlib

User Manual

Version 1.1.0
21 August 2002

Jürgen Knödlseder
Centre d'Etude Spatiale des Rayonnements
knodlseder@cesr.fr
http://www.cesr.fr./~jurgen/index.html

Note to the user

This software has been written to analyse data of the SPI telescope onboard INTEGRAL. Particular care has been taken in making the software user friendly and well documented. If you appreciated this effort, and if this software and User Manual were useful for your scientific work, the author would appreciate a corresponding acknowledgment in your published work.

Contents

1 Introduction

The executable `spi_psd_fitlib` performs library fitting of PSD curve data. This task is complementary to the executable `dp_spi_psd` which performs routinely pulse fitting of PSD curves on science window basis. `spi_psd_fitlib` is aimed to allow for independent pulse fitting, in particular in view of a deeper investigation of PSD pulse fitting performances. One may use `spi_psd_fitlib` to compare library fitting results for different PSD template libraries or different PSD algorithm parameters.

For each curve in the input data structure `spi_psd_fitlib` determines the best fitting pair of library templates by means of a least Chi-squared fitting technique (the algorithm is an exact copy of the PSD onboard template fitting algorithm). The fitting result consists of

- `PSD_FIT_TTP1`: The library template time-to-peak value of the smaller of the two fitted templates.
- `PSD_FIT_TTP2`: The library template time-to-peak value of the larger of the two fitted templates.
- `PSD_FIT_AMP`: The relative amplitude of the smaller library template with respect to the larger template. By definition, `PSD_FIT_AMP` ≤ 0.5.
- `PSD_FIT_CHISQR`: The Chi-squared of the fitting result.

These fitting result are automatically inserted in the corresponding columns in the curve `OSM` data structures. Previous fitting results that may already be present in these columns will **be replaced**.

The `spi_psd_fitlib` fitting algorithm is identical to the PSD onboard algorithm, hence the results may be compared to the onboard fitting results. The only difference between onboard and onground fitting is the length of the pulse shape, which consists of 90 time-steps onboard from which the first 79 are only transmitted to ground. However, a reasonable PSD pulse shape should be much shorter (typically 30 - 40 time steps), and the fitting result should not be much affected by this difference.

The executable `spi_psd_fitlib` is written in the ANSI C++ language. It has been developped under ISDC support platform 4.1 and requires `spi_psdlib` 1.6.0 and `spi_toolslib` 1.8.0 or higher.

2 Getting started

Before installing `spi_psd_fitlib`, make sure that the ISDC support platform 4.1 or higher is installed on your system, and that you have installed the libraries `spi_psdlib` 1.6.0 and `spi_toolslib` 1.8.0 or higher.

After downloading the `spi_psd_fitlib.tar.gz` file, step into a directory that should hold the distribution, move the `spi_psd_fitlib.tar.gz` file into this directory and type:

```
$ gunzip spi_psd_fitlib.tar.gz
$ tar xvf spi_psd_fitlib.tar
```

The first command uncompresses the distribution file, the second unpacks the files.

Before configuration, the distribution needs to be reset to a clean state. To do this, type

```
$ make distclean
```

Then, configure the distribution. It is assumed here that you have previously installed the ISDC support platform, thus you should type

```
$ ~/bin/ac_stuff/configure
```

Finally, build the distribution by typing

```
$ make global_install
```

To perform a unit test, type

```
$ make test
```

Make sure that the test data `spi_test_data-1.0.tar.gz` are available at your site (they should reside in a directory whose name is defined by the `ISDC_TEST_DATA_DIR` environment variable).

3 Parameter file

```
#############################################################################
#                                                                           #
#               Centre d'Etude Spatiale des Rayonnements                    #
#                     (in collaboration with ISDC)                          #
#                                                                           #
#                          SPI PSD ANALYSIS                                 #
#                                                                           #
# ------------------------------------------------------------------------- #
#                                                                           #
#  File:      spi_psd_fitlib.par                                            #
#  Version:   1.1.0                                                         #
#  Component: PA                                                            #
#                                                                           #
#  Author:    Juergen Knoedlseder                                           #
#             knodlseder@cesr.fr                                            #
#             CESR                                                          #
#                                                                           #
#  Purpose:   Parameter file of the SPI PSD library fitting                 #
#             executable                                                    #
#                                                                           #
#  History:   1.1.0  21-Aug-2002  First ISDC delivery (Rev. 1)              #
#                                                                           #
#############################################################################
#
# The input DOLs
#===============
curveDOL,  s,ql, "og_spi.fits[1]",,,"Input DOL (SWG/IDX/OG)"
libDOL,    s,ql, "psd.fits[SPI.-LIB.-PSD]",,,"PSD template library DOL"
algoparDOL,s,ql, "psd.fits[SPI.-ALGO-PSD]",,,"PSD algorithm parameters DOL"
#
# Task parameters
#================
set,        i,ql, 0,0,1, "Template library set (0/1)"
onground,   b,ql, yes,,, "On-ground pulse filtering ?"
fitalltpls, b,ql, yes,,, "Use all available templates for fitting ?"
adc0_gain,  r,h,  1.0,,, "Gain PSD ADC 0"
adc1_gain,  r,h,  1.0,,, "Gain PSD ADC 1"
adc2_gain,  r,h,  1.0,,, "Gain PSD ADC 2"
adc3_gain,  r,h,  1.0,,, "Gain PSD ADC 3"
adc0_offset,r,h,  0.0,,, "Offset PSD ADC 0"
adc1_offset,r,h,  0.0,,, "Offset PSD ADC 1"
adc2_offset,r,h,  0.0,,, "Offset PSD ADC 2"
adc3_offset,r,h,  0.0,,, "Offset PSD ADC 3"
#
# Standard parameters
#====================
clobber,b,h, no,  ,, "Overwrite existing data structures ?"
```

The following parameters have to be specified:

- `curveDOL` specifies the input group that holds the PSD curves that should be fitted. The input group

may be either a single science window or an observation group.

- `libdol` specifies the DOL of the template library that should be used for pulse fitting.
- `algopardol` specifies the DOL of the PSD algorithm parameters that should be used for pulse fitting.
- `set` specifies the template set from that should be used for the pulse fitting.
- `onground` specifies if the PSD pulses should be filtered onground using the criteria defined in the algorithm parameters.
- `fitalltpls` specifies if all templates from the library should be used for fitting, or if only the number of templates specified in the PSD algorithm parameter block should be used (default is `yes`).
- `adc0_gain` specifies the gain correction that should be applied for the PSD ADC convertor 0. As default select 1.0.
- `adc1_gain` specifies the gain correction that should be applied for the PSD ADC convertor 1. As default select 1.0.
- `adc2_gain` specifies the gain correction that should be applied for the PSD ADC convertor 2. As default select 1.0.
- `adc3_gain` specifies the gain correction that should be applied for the PSD ADC convertor 3. As default select 1.0.
- `adc0_offset` specifies the offset correction that should be applied for the PSD ADC convertor 0. As default select 0.0.
- `adc1_offset` specifies the offset correction that should be applied for the PSD ADC convertor 1. As default select 0.0.
- `adc2_offset` specifies the offset correction that should be applied for the PSD ADC convertor 2. As default select 0.0.
- `adc3_offset` specifies the offset correction that should be applied for the PSD ADC convertor 3. As default select 0.0.
- `clobber` specifies if existing data structures should be overwritten.

4 Interface definition

`spi_psd_fitlib` works either on indivdual science windows or on observation group. On output, the columns `PSD_FIT_TTP1`, `PSD_FIT_TTP2`, `PSD_FIT_AMP`, and `PSD_FIT_CHISQR` of the `PRP` data structure are updated with the fitting results. Any previously existing result is overwritten.

5 Error codes

The following error codes are defined:

```
SPI_PSD_FITLIB_ERROR_MEM_ALLOC        -230600
```

They have the following meaning:

- `SPI_PSD_FITLIB_ERROR_MEM_ALLOC` : an error occured during dynamical memory allocation. Probably your system resources are not sufficient to run `spi_psd_fitlib`.