spiskymax User Manual and Explanatory Supplement

A. Strong
MPE Garching

Issue 3 September 18, 2002

Contents

Chapter 1

Introduction

1.1 Scope

The coded-mask imaging γ-ray spectrometer SPI on the INTEGRAL space observatory will detect point sources and diffuse extended emission with an angular accuracy of about 1° over its energy range of 40–8000 keV. The purpose of the software package described here is to represent the measurement in terms of pixelized models on the sky, including estimates of the uncertainty. This tool is oriented towards large-scale surveys (eg. GCDE), which combine a large set of individual pointings of the spacecraft. It concentrates on spatial as opposed to spectral information, although (through using the different energy channels of the measurement) it will be an important method to generate spectra of diffuse emission.

The principal use will be generating maps in lines and continuum. γ-ray emission. Examples are the 1809 keV ^{26}Al line and diffuse continuum γ-ray s. The analysis of measurements with this tool is complementary to image model fitting (e.g. from *spidiffit*) and methods specifically designed for point sources (e.g. *spiros*).

The algorithm performs fitting of raw data (binned counts for many observations) to a pixelated model using the full instrument response information. In addition to parameter estimation, a Bayesian statistical analysis is used to obtain profiles (integrated intensities) with error estimates.

The package is referred to as *spiskymax* . The package is closely related to other imaging packages: response, convolution, background treatments are common or similar.

A presentation of *spiskymax* with example simulations including the Galactic Deep Exposure (GCDE) can be found at

http://www.gamma.mpe-garching.mpg.de/∼aws/integral.html
This manual describes *spiskymax* Version 26.0.

1.2 References and Links

This is not a complete reference list, but the cited papers and links contain very extensive references.

1. MAXENT5 User's Manual: http://www.maxent.co.uk (under 'documents')

2. Gull, S.F. (1989) Developments in maximum entropy data analysis, in 'Maximum entropy and Bayesian methods', ed. J. Skilling, Kluwer, Dordrecht

3. Skilling, J. (1989) Classic maximum entropy, in 'Maximum entropy and Bayesian methods', ed. J. Skilling, Kluwer, Dordrecht

4. Sivia, D.S. Data Analysis: A Bayesian Tutorial, Oxford University Press, 1997, ISBN 0 19 851889 7 Very useful self-contained reference, paperback and easily obtained.

5. http://www.gamma.mpe-garching.mpg.de/∼aws/integral.html

   presentations of simulations and calibration data analysed with *spiskymax*

6. Cambridge MCMC Preprint service : http://www.statslab.cam.ac.uk/ mcmc/

1.3 Document History

1. 2 Mar 2002: Version 2: spiskymax Version 20.0

2. 18 Sept 2002: Version 3: spiskymax Version 26.0, with addition of GTI and PSD datasets

Chapter 2

The *spiskymax* algorithm

2.1 Introduction

The aim of *spiskymax* is to generate images from SPI data. SPI is a coded-mask instrument, so direct deconvolution is in principle possible, but in practice the response is complex and the data include many pointing directions of the instrument, so 'indirect imaging' is essential. The situation is similar to a radio inferometer telescope, where Fourier inversion is in principle possible but incomplete and noisy data mean that methods like Richardson-Lucy or 'clean' are used. Indirect imaging implies 'forward-folding': for any candidate image we convolve with the instrument response function and compare the result with the observed data. This gives the basis for any iterative method which seeks to successively improve the agreement of the predicted with the observed data by adjusting the image. One approach is to regard the image as made up of 'point sources' and to adjust their positions and fluxes to give a best-fit to the data. This is the basic principle of the *spiros* method. Other methods extend the point-source idea to more complex objects like wavelets or spline functions and fit the data to a combination of these 'pixons'. These methods have relatively few basis parameters and so can be regarded as essentially 'model-fitting'.

spiskymax on the other hand regards the image as a pixelated skymap, and the aim is to obtain the intensity in each pixel. The most important idea is that the data constrain the image within some limits in an N-dimensional space, where N is the number of pixels. Hence there is no unique 'best' image and we have to make some choice out of all the possible images within the constrained region. Maximum Entropy as implemented here is one way of doing this, and of quantifing the uncertainty of the result. Since the number of pixels is usually large (e.g. 10^4-10^6) the techniques involved are rather different from those of model-fitting.

2.2 Maximum Entropy Method

The Maximum Entropy method (MaxEnt) is a general technique for deconvolution of data, which has been developed over the past 20 years with special emphasis on applications in spectrosopy and imaging.

The implementation in *spiskymax* is based on the MEMSYS5 package which uses advanced numerical techniques to perform the computations, which are hard on account of the large dimensionality of the problem. The standard papers on the method are Gull (1989) and Skilling (1989). A good exposition of the principles can be found in Sivia (1997) as well as the literature associated with the MEMSYS5 package itself (see 'References and Links'). The original 'maximum entropy method' as applied to imaging was based on the idea of smoothness, the idea being to obtain the 'flattest image consistent with the data', where 'flatness' is measured by the entropy defined as the sum over pixels $S = -\sum_i p_i ln p_i$, where p_i is the fraction (proportion) of the image in pixel i. It can be shown that this form of S is the only one possessing the necessary properties like consistency under rebinning, partitioning etc. These arguments are given in detail in the literature, but the basic principle is to generate a 'conservative' solution which contains 'only structure for which there is evidence in the data'. This approach, though very fruitful, contained some logical problems (e.g. how to define 'consistency with the data', how to quantify the uncertainties). The further developments leading to MEMSYS5 came from presenting the method as a particular application of Bayesian statistics, with entropy providing the basis for the 'prior probabilities'. This 'Quantified Maximum Entropy' or 'Classic Maximum Entropy' is based on the concept of the posterior probability distribution of the full N-dimensional space of image pixels, and this allows explicit computation of uncertainties and a well-defined criterion for the 'best' image.

MaxEnt is described in detail in the MEMSYS5 User's Manual (to be found at http://www.maxent.co.uk), and the reader is referred to this for an extensive exposition. The notation in the present document follows as far as possible the MEMSYS5 User's Manual.

2.3 MaxEnt Formulae

We give here a brief summary of the main formulae involved, omitting technicalities. The posterior probability of the image I given the data D is $P(I|D) \propto e^{-L+\alpha S}$ where $L = -lnP(D|I)$ is (minus) the log-likelihood function and $S = \sum_i (I_i - m_i - I_i ln\frac{I_i}{m_i})$ is the entropy in the form appropriate to 'positive additive distributions'. Here m_i is the default value to which a pixel will be assigned in the absence of constraints from the data (S is maximum at $I = m$). α, which determines the balance between the influence of the prior and the data on the result, is in principle unknown; however in Classical MaxEnt α can itself be treated by Bayesian methods and ideally 'marginalized' out, in practice a best value is determined by maximizing the probability of the data P(D). It can be shown that the best value corresponds to equating the amount of structure in the image with the 'number of good measurements' in the data. In cases where this estimate of α is not adequate, an option to treat it as a user-defined parameter can be invoked. m is generally taken to be a 'flat' image with a value approximately equal to the mean expects intensity; the exact value is not critical. (NB An option to allow user-defined maps defining m based on physical models would be desirable in future.)

The MEMSYS5 package performs an iterative search to obtain $P(I|D)$. To make the entire method tractable, $P(I|D)$ is represented as an N-dimensional Gaussian. This can be used to compute 'error bars' on any linear combination of the pixels, allowing profiles across the image or fluxes of 'point sources' with their associated uncertainties to be generated.

2.4 Application to Gamma-ray Astronomy

The MaxEnt method has been applied extensively to the CGRO/COMPTEL Compton Telescope data (refs), in particular was used to generate the ^{26}Al 1.8 MeV and continuum full-sky maps. It has also been used for CGRO/EGRET data (ref).The experience gained from COMPTEL was important for the SPI application. A straightforward application of MEMSYS5 to gamma-ray data is in fact not difficult, the main problem as usual in gamma-ray astronomy being the instrumental background which is both large and in general unknown. The approach taken in *spiskymax* is to fit the background along with the image, in fact to treat the background as an extension of the image itself. This is quite natural since the image is simply a set of parameters (intensity in each pixel) and the background is another set (background per detector and pointing). Since the entropy of the parameters defining the background should not be allowed to influence the image, a scaling factor is included which can be invoked to reduce to a negligible level the contribution of these parameters to S.

2.5 Application to SPI

spiskymax is adapted to the particular needs of SPI. The background temporal variations are treated via a template (data type SPI-BMOD-DSP-IDX) prepared by *spiback*, and the coefficients are fitted during the imaging process. The input count spectra (SPI.-OBS-DSP) contain data for many energy bins, and *spiskymax* will analyse a subset (or all) of these energies as specified by the parameters *energy_range_min*, *energy_range_max*. The number of iterations can be specified since the automatic stopping criterion is not always appropriate and in any case may not be reached in the CPU time available. Sources to be analyzed are defined by their positions in the input source catalogue (SPI.-SRCL-CAT); only those with the SEL_FLAG flag =1 are analyzed, and their fluxes and 1σ errors written to the output catalogue.

2.6 Image and Data space, likelihood function

We distinguish image space and data space in the usual way, and define the instrument response as the relation between them. The image is I_j and the expected data is d_k. The expected background is b_k Let R_{jk} be the response of data element k to image element j. Then

$$d_k = \sum_j R_{jk} I_j + b_k$$

Similarly the background can be constructed from components of a background model B_{ik}

$$b_k = \sum_i \theta_i B_{ik}$$

where θ_i now introduces background parameters. In this way we can treat image and background model in the same way in the subsequent analysis, and θ_i includes both. The only formal difference between image and background model is that the image is convolved with R_{jk} and the background is not:

$$d_k = \sum_j R_{jk} I_j + \sum_{i=N_I+1}^{i=N_I+N_B} \theta_i B_{ik}$$

where there are N_I image components and N_B background components, and $N = N_I + N_B$.

The likelihood function is:

$$P(D|I) = \prod_k e^{-d_k} d_k^{n_k} / n_k!$$

where n_k are the measured data (denoted collectively by D).

2.7 *spiskymax* Output

spiskymax outputs four main types of information: diagnostics, skymaps, profiles, source fluxes. In addition extensive 'debug' information may be output to help tracing problems. The diagnostics allow the user to judge the progress of the iterative procedure and evaluate the analysis; they are output as simple ASCII. The skymaps and profiles are output to FITS files with the ISDC template SPI.-SKY.-IMA, with an additional FITS file for the index with template SPI.-SKY.-IMA-IDX. The source fluxes and errors are output to FITS files with template SPI.-SRCL-RES.

2.7.1 Image output

All maps and profiles are written as FITS IMAGE extensions to the SPI.-SKY.-IMA file. Each extension has its header defining the coordinates relevant to the map or profile so that they can be read by standard FITS tools like fv. To assist, a sample plotting program in *idl* is provided in the delivery.

The sequence is:

```
skymap energy range 1

skymap energy range 2
....
skymap energy range n


profile 1
```

```
   energy range 1
    flux
    flux error
   energy range 2
    flux
    flux error
....
   energy range n
    flux
    flux error


profile 2
   energy range 1
    flux
    flux error
   energy range 2
    flux
    flux error
....
   energy range n
    flux
    flux error


etc.
```

2.7.2 Source catalogue output

This is in standard ISDC source catalogue format, with the fields for source ID and position copied from the input catalogue, and the fields for source flux and flux error filled from the *spiskymax* analysis. Only sources flagged with SEL_FLAG = 1 in the input catalogue are processed and output.

2.7.3 Diagnostic output

The diagnostics are output directly from the MEMSYS5 package and for technical reasons go directly to the standard output (screen) unless piped to a file (e.g. ‘spiskymax ¿ myoutput’). The results of the source analysis are also output to the log file (as well as to the source catalogue). The following is the output produced from the parameter file described in the example section '3C273' including the output to the log file.

```
Log_1  : CommonPreparePARsStrings status=0
Log_1  :  spiskymax_get_par ends
Log_1  : >> spiskymax_processing begins
Log_1  : read_SPI_OBS_DSP begins
Log_1  : read_SPI_OBS_DSP ends
Log_1  : DALobjectFindElement status=0
Log_1  : DAL3GENindexGetNumMembers status=0
Log_1  : DAL3GENindexGetNumMembers NumMembers=1
Log_1  : DAL3GENindexGetMember status=0
Log_1  : image_definition begins
Log_1  : image_definition ends
Log_1  :        +++++++++       memsys5 analysis for energy range #1 +++++++++++++++
Log_1  :  Iteration    1

  Entropy ===   0.0000e+00    Test === 0.0000    Chisq ===  3.2388e+03
  LogProb ===  -4.0100e+03    Code ===   0       Good  ===  0.0000e+00
  Omega   ===   0.003160      dist === 0.4316    Alpha ===  5.6818e+01
  Ntrans  ===      7                             Code  ===  001010
Log_1  :  Iteration    2

  Entropy ===  -1.1716e+01    Test === 0.6501    Chisq ===  2.1016e+03
  LogProb ===  -4.1127e+03    Code ===   0       Good  ===  6.7147e+00
  Omega   ===   0.005044      dist === 0.2495    Alpha ===  5.6818e+01
  Ntrans  ===     18                             Code  ===  010011
Log_1  :  Iteration    3

  Entropy ===  -4.6802e+00    Test === 0.0763    Chisq ===  1.7072e+03
  LogProb ===  -3.5146e+03    Code ===   0       Good  ===  5.6983e+00
  Omega   ===   0.010714      dist === 0.4428    Alpha ===  2.2134e+01
  Ntrans  ===     30                             Code  ===  001011
Log_1  :  Iteration    4

  Entropy ===  -7.8380e+01    Test === 1.2622    Chisq ===  8.3256e+03
  LogProb ===  -8.2990e+03    Code ===   0       Good  ===  9.1382e+00
  Omega   ===   0.002634      dist === 0.4546    Alpha ===  2.2134e+01
  Ntrans  ===     40                             Code  ===  010011
```

```
Log_1  :  Iteration    5

  Entropy ===  -2.9232e+01    Test === 0.4026    Chisq ===  8.3350e+02
  LogProb ===  -3.4630e+03    Code ===   0       Good  ===  7.8946e+00
  Omega   ===   0.006101      dist === 0.1344    Alpha ===  1.5306e+01
  Ntrans  ===     51                             Code  ===  001011
Log_1  :  Iteration    6

  Entropy ===  -1.9820e+01    Test === 0.0666    Chisq ===  7.2164e+02
  LogProb ===  -3.0641e+03    Code ===   0       Good  ===  8.7264e+00
  Omega   ===   0.014383      dist === 0.1681    Alpha ===  9.9552e+00
  Ntrans  ===     65                             Code  ===  001011
Log_1  :  Iteration    7

  Entropy ===  -2.5671e+01    Test === 0.0134    Chisq ===  5.9182e+02
  LogProb ===  -2.9530e+03    Code ===   0       Good  ===  9.6517e+00
  Omega   ===   0.018883      dist === 0.0998    Alpha ===  6.8541e+00
  Ntrans  ===     78                             Code  ===  001011
Log_1  :  Iteration    8

  Entropy ===  -2.7975e+01    Test === 0.0103    Chisq ===  5.4891e+02
  LogProb ===  -2.8691e+03    Code ===   0       Good  ===  1.0833e+01
  Omega   ===   0.028250      dist === 0.1322    Alpha ===  4.8103e+00
  Ntrans  ===     93                             Code  ===  001011
Log_1  :  Iteration    9

  Entropy ===  -3.1926e+01    Test === 0.0043    Chisq ===  5.0671e+02
  LogProb ===  -2.8113e+03    Code ===   0       Good  ===  1.2034e+01
  Omega   ===   0.039180      dist === 0.1034    Alpha ===  3.4599e+00
  Ntrans  ===    108                             Code  ===  001011
Log_1  :  Iteration   10

  Entropy ===  -3.4267e+01    Test === 0.0015    Chisq ===  4.8698e+02
  LogProb ===  -2.7679e+03    Code ===   0       Good  ===  1.3224e+01
  Omega   ===   0.055768      dist === 0.1001    Alpha ===  2.5213e+00
  Ntrans  ===    123                             Code  ===  001011
Log_1  :  Iteration   11

  Entropy ===  -3.6729e+01    Test === 0.0013    Chisq ===  4.7296e+02
  LogProb ===  -2.7364e+03    Code ===   0       Good  ===  1.4298e+01
```

```
  Omega   ===   0.077197      dist === 0.1016    Alpha ===  1.8588e+00
  Ntrans  ===    140                             Code  ===  001010
Log_1  :  Iteration   12

  Entropy ===  -3.9044e+01    Test === 0.0014    Chisq ===  4.6339e+02
  LogProb ===  -2.7133e+03    Code ===   0       Good  ===  1.5878e+01
  Omega   ===   0.109387      dist === 0.1198    Alpha ===  1.3835e+00
  Ntrans  ===    160                             Code  ===  001010
Log_1  :  Iteration   13

  Entropy ===  -4.1537e+01    Test === 0.0016    Chisq ===  4.5620e+02
  LogProb ===  -2.6969e+03    Code ===   0       Good  ===  1.8390e+01
  Omega   ===   0.160000      dist === 0.1208    Alpha ===  1.0382e+00
  Ntrans  ===    183                             Code  ===  001010
Log_1  :  Iteration   14

  Entropy ===  -4.4061e+01    Test === 0.0016    Chisq ===  4.5049e+02
  LogProb ===  -2.6846e+03    Code ===   0       Good  ===  1.9866e+01
  Omega   ===   0.217148      dist === 0.1296    Alpha ===  7.8474e-01
  Ntrans  ===    208                             Code  ===  001010
Log_1  :  Iteration   15

  Entropy ===  -4.6926e+01    Test === 0.0018    Chisq ===  4.4578e+02
  LogProb ===  -2.6761e+03    Code ===   0       Good  ===  2.2356e+01
  Omega   ===   0.303542      dist === 0.1351    Alpha ===  5.9726e-01
  Ntrans  ===    234                             Code  ===  001010
Log_1  :  Iteration   16

  Entropy ===  -5.0040e+01    Test === 0.0018    Chisq ===  4.4182e+02
  LogProb ===  -2.6701e+03    Code ===   0       Good  ===  2.4618e+01
  Omega   ===   0.411860      dist === 0.1363    Alpha ===  4.5745e-01
  Ntrans  ===    260                             Code  ===  001010
Log_1  :  Iteration   17

  Entropy ===  -5.3244e+01    Test === 0.0015    Chisq ===  4.3851e+02
  LogProb ===  -2.6657e+03    Code ===   0       Good  ===  2.5745e+01
  Omega   ===   0.528497      dist === 0.1337    Alpha ===  3.5232e-01
  Ntrans  ===    287                             Code  ===  001010
Log_1  :  Iteration   18
```

```
  Entropy ===  -5.6472e+01    Test === 0.0011    Chisq ===  4.3579e+02
  LogProb ===  -2.6629e+03    Code ===   0       Good  ===  2.8029e+01
  Omega   ===   0.704359      dist === 0.1407    Alpha ===  2.7266e-01
  Ntrans  ===    315                             Code  ===  001010
Log_1  :  Iteration   19

  Entropy ===  -6.0165e+01    Test === 0.0009    Chisq ===  4.3356e+02
  LogProb ===  -2.6613e+03    Code ===   0       Good  ===  2.9639e+01
  Omega   ===   0.903395      dist === 0.0351    Alpha ===  2.4667e-01
  Ntrans  ===    346                             Code  ===  001000
Log_1  :  Iteration   20

  Entropy ===  -5.9494e+01    Test === 0.0001    Chisq ===  4.3327e+02
  LogProb ===  -2.6604e+03    Code ===   0       Good  ===  3.0358e+01
  Omega   ===   1.034315      dist === 0.0398    Alpha ===  2.5276e-01
  Ntrans  ===    380                             Code  ===  001000
Log_1  :  Iteration   21

  Entropy ===  -6.0327e+01    Test === 0.0000    Chisq ===  4.3281e+02
  LogProb ===  -2.6603e+03    Code ===   0       Good  ===  2.9854e+01
  Omega   ===   0.978911      dist === 0.0050    Alpha ===  2.5276e-01
  Ntrans  ===    415                             Code  ===  000000
Log_1  :  +++++++++Statistics+++++++++
  Entropy ===  -6.0286e+01    Test === 0.0000    Chisq ===  4.3283e+02
  LogProb ===  -2.6602e+03    Code ===   0       Good  ===  2.9783e+01
  Omega   ===   0.977243                         Alpha ===  2.5276e-01
  Ntrans  ===    435                             Code  ===  000000
Log_1  :  Good     =       29.8 = [       28.6 ,       30.9 ]

Log_1  :  Evidence =     -2660.2 = [     -2660.5 ,     -2660.0 ]  = Log[e]Prob(Data)

Log_1  :  ------------------- source fluxes and error analysis -----------------
Log_1  :  ------------------- source flux units: photons cm^-2 s^-1 ------------
Log_1  : === Testing for source 1 at   0.00   0.00 ON radius=  0.50 OFF radius=  2.00 : flux=4.396e-04 +-4.760e-05 sigma=   9.24
Log_1  :  ------------ end of source fluxes and error analysis -----------------
Log_1  :
 ------------ Analysing source catalogue for energy range 100.00-200.00 keV

Log_1  : total number of sources in  input catalogue=5
Log_1  : total number of sources in output catalogue=1
```

```
Log_1  : === Testing for source 1 at   0.00   0.00 ON radius=  0.50 OFF radius=  2.00 :
             flux=4.396e-04 +-4.760e-05 sigma=   9.24
Log_1  :
 ----------End analysing source catalogue for energy range 100.00-200.00 keV

Log_1  :         +++++++++ end of memsys5 analysis for energy range #1 ++++++++++++++++
Log_1  :         +++++++++        memsys5 analysis for energy range #2 ++++++++++++++++
Log_1  :  Iteration    1
```

...and so on for all energy ranges.

This run illustrates how the value of Chisq (defined as -2 ln L, see 'MaxEnt Formulae') decreases with iteration as Alpha (α) decreases, so that the influence of the entropy is reduced and the fit to the data improves. At the same time LogProb (ln P(D)) increases until the stopping criterion Omega =1 is satisfied. The diagnostics can best be understood by reference to the sections in this document describing the algorithm, but for a proper understanding we recommend consulting the MEMSYS5 User's Manual which gives full explanations and examples.

The source analysis for the first energy range is shown at the end of this output; the source is detected at 9.24σ with flux 4.396 ± 0.476 10^{-4} cm^{-2} s^{-1} .

2.7.4 Debug output

There are many reasons why a run can fail, for example the input data may be inconsistent or missing. In these cases setting debug = 1 will printout information to assist in understanding the problem, including the input data. Setting debug =2 prints out even more and is only necessary when the problem lies deeper. debug = 1 also invokes a test of the correct functioning of the convolution routines which can be useful when the program runs but gives unexpected results related e.g. to an incorrect IRF.

Chapter 3

Software Specification

3.1 Software Requirements

3.1.1 Functions and Controls

spiskymax uses SPI measured data to generate intensity images. In detail, this encompasses the functions of:

1. Obtain data
2. Obtain instrument response
3. Obtain instrumental background model
4. Perform maximum entropy analysis
5. Generate profiles with error estimates
6. Generate source fluxes with error estimates
7. Output maps, profiles and source catalogue

These functions are controlled in detail through user settings of the parameters:

1. Data range selections
2. etc TBW

3.1.2 Interfaces

Input

1. SPI binned event data (as produced by spihist)
2. SPI pointing data
3. SPI energy bounds data
4. SPI dead times data
5. SPI good times data (used only to obtain detector IDs)
6. SPI background model
7. SPI response (IRF)
8. SPI PSD efficiencies
9. SPI PSD reponse
10. source catalogue
11. fitting parameters specification

Output

1. maximum entropy skymaps as FITS extensions in SPI.-SKY.-IMA
2. intensity profiles as FITS extensions in SPI.-SKY.-IMA
3. source catalogue as SPI.-SRCL-CAT

3.2 Software Architecture and Design

spidiffit is implemented in independent sub-modules, which can be tested separately:

- Prepare data, models, and response
- Perform maximum entropy analysis
- Prepare outputs and results

3.3 Development and Testing

3.3.1 Relation to other SPI software

spiskymax has much in common with other SPI programs. It uses the same data and response function R_{jk} as $spidiffit$ and $spiskycnv$.

3.3.2 Development Plan

3.3.3 Testing

Standard test cases are defined to measure the science validity and performance of the package. These are based on simulated data for the cases of:

- GCDE exposure pattern fit in 5 spectral bands
- Galactic Centre Deep Exposure in 1809 keV line band

3.3.4 Performance

To be written.

Chapter 4

Examples

The examples illustrate how to run *spiskymax* for various typical cases. They refer to spiskymax Version 20.0; later versions have some additional parameters (see Section 'Help File' for latest version and list of changes) but the principles are identical. The data is either simulated with ISDC Instrument Specific Software (ISSW): gensky, spiskycnv, spisimprep, spidead, spiback or comes from the Bruyeres-le-Chatel calibration and was generated with spihist. The full datasets and spiskymax.par files can be found at TBD. For the first example the parameters are discussed in a 'tutorial' fashion.

4.1 '3C273'

This uses a standard $5^o \times 5^o$ dither pattern with total time 10^6 s centred on the source, flux = 0.2 $(\mathrm{E/keV})^{-2}$ cm^{-2} s^{-1} keV^{-1} in the 400-1000 keV band. This is a very rough approximation to the spectrum of 3C273 based on COMPTEL results and is chosen for illustration only. The input datasets are the detector spectra, pointing, deadtime, background and source catalogue. The output is the image and any required profiles, and a source catalogue with fluxes of the sources selected in the input catalogue. The image coordinates and pixel size are specified with *image-fov* = SURVEY which means that the range of (l,b) or (α, δ) are explicitly given via *chi_0, chi_1, d_chi, psi_0, psi_1, d_psi* in degrees. (In this case *centre-long* etc. are not used). In this example the source is centred on (0,0) rather than the actual celestial position of 3C273. No profiles are requested so *n_profiles* =0. The source flux from the input catalogue is derived using *source_ON_radius_1*=2.0^o and *source_OFF_radius_1*=5.0^o using the method described in the chapter 'Help file'.

The spiskymax.par file is as follows:

```
##############################################################################
#
# =#= spiskymax 20.0 =#=
# ~aws/integral/spectrometer/simulations/sources.3

debug  ,i,h,0,0,2,"0=silent,1,2 gives more verbose output"
display,i,h,0,0,2,"1,2 displays input sky using root "
title  ,s,h,"3C273, 400-1000 keV, 10^6 sec  ",,,"title to write as comment to FITS header"
rogroup,s,h,"",,,"xxx R/O Group"
rwgroup,s,h,"",,,"xxx R/W Group"

counts_input_file,s,h,"SPI-OBS-DSP.spiskycnv_14.fits[1]",,,"xxx input count  file"
pointing_input_file,s,h,"SPI-OBS-PNT.spisimprep_2.fits[1]",,,"xxx input pointing  file"
ebounds_input_file, s,h,"SPI-EBDS-SET.spiskycnv_14.fits[1]",,,"xxx input energy bounds  file"
deadtime-dol,s,h,"SPI-OBS-DTI.spisimprep_2.fits[1]",,,"xxx DTI deadtime/livetime input file"
background_input_file,s,h,"SPI-BMOD-DSP-IDX.spiback.fits[1]",,,"xxx input background file"
irf_input_file,s,h,"/afs/ipp/mpe/gamma/integral/isdc/data/spi/IRFs/spi_irf_grp_0007.fits[GROUPING]" ,,," input IRF file 85 detectors"
source-cat-dol,s,h,"SPI-SRCL-CAT.test_catalogue.fits[1]",,,"input catalogue of sources "
image-idx,s,h , "SPI-SKY-IMA-IDX.spiskymax_19_back1.fits(SPI.-SKY.-IMA-IDX.tpl)",,,"output skymap image index file"
image-int,s,h ,     "SPI-SKY-IMA.spiskymax_19_back1.fits",,,"    output skymap images file"
source-res-dol,s,h,"SPI-SRCL-RES.spiskymax_19_back1.fits(SPI.-SRCL-RES.tpl)",,,"output catalogue of sources "


skymap_system,s,h,G,,,"input skymap coordinate system C=celestial, G=Galactic"
image-fov,s,h,"SURVEY",,,"Image field of view: SURVEY  USER  POINTING POINTING+FCFOV POINTING+ZCFOV POINTING-CENTER"
center-long,r,h, 1.0,,,"Longitude of centre pixel (degrees) [image-fov= USER]"
center-lat ,r,h, 2.0,,,"Latitude  of centre pixel (degrees) [image-fov= USER]"
image-pixel-long,r,h,0.5,,,"Longitude pixel size     (degrees) [image-fov= USER,POINTING*]"
image-pixel-lat,r,h,0.5,,,"Latitude  pixel size      (degrees) [image-fov= USER,POINTING*]"
image-dim-long,i,h,40,,,"Longitude dimension        (pixels) [image-fov= USER,POINTING-CENTER]"
image-dim-lat ,i,h,41,,,"Latitude  dimension        (pixels) [image-fov= USER,POINTING-CENTER]"

chi_0, r,h, -20.0,,,"Longitude of first pixel (degrees)[image-fov= SURVEY]"
chi_1, r,h, +20.0,,,"Longitude of last  pixel (degrees)[image-fov= SURVEY]"
d_chi, r,h,   0.2,,,"Longitude binsize        (degrees)[image-fov= SURVEY]"
psi_0, r,h, -20.0,,,"Latitude  of first pixel (degrees)[image-fov= SURVEY]"
psi_1, r,h, +20.0,,,"Latitude  of last  pixel (degrees)[image-fov= SURVEY]"
d_psi, r,h,   0.2,,,"Latitude  binsize        (degrees)[image-fov= SURVEY]"
```

```
energy_range_min,i,h,  1,,,"minimum energy range sequence number as in ebounds file: 1,2,3..."
energy_range_max,i,h,  1,,,"maximum energy range sequence number as in ebounds file: 1,2,3..."

max_iter,i,h , 30,0,1000,"maximum number of memsys5 iterations"
image_default, r,h,1.0e-3,,,"image default intensity, cm-2 sr-1 s-1"
background_scaling_default,r,h,1.0e-0,,,"background scaling default"
background_entropy_factor ,r,h,1.0e-0,,,"background factor to reduce contribution to entropy"
background_method,i,h,1,1,2,"background method:1=free 2=fixed"

memsys_iBayes,i,h,1,1,3,"memsys Bayesian method code: 1= classic,  3=ad hoc specified by fAim    "
memsys_iEntropy,i,h,1,1,5,"memsys entropy type:         1= standard, 5=fixed total=sum over defaults"
memsys_fAim,r,h,1.0e-00,1e-10,1e10,"memsys stopping criterion: classic: 1/Omega =1.0, ad hoc: alpha "

n_source_fluxes,i,h,1,0,100,"number of source fluxes with errors"
source_chi_1,r,h,0.0,,,"longitude (degrees) of source 1"
source_psi_1,r,h,0.0,,,"latitude  (degrees) of source 1"
source_ON_radius_1, r,  h, 2.00 ,,,"ON radius (degrees)  for  source 1"
source_OFF_radius_1,r,h, 5.00,,,"OFF radius (degrees) for  source 1"

n_profiles,i,h,0,0,100,"number of intensity profiles with errors"
profile_chi_0,s,h,"-20.0",,,"Longitude of first pixel of profiles (degrees)"
profile_chi_1,s,h,"+20.0",,,"Longitude of  last pixel of profiles (degrees)"
profile_d_chi,s,h,"  2.0",,,"Longitude binsize        of profiles (degrees)"
profile_psi_0,s,h," -0.0",,," Latitude of first pixel of profiles (degrees)"
profile_psi_1,s,h," +0.0",,," Latitude of  last pixel of profiles (degrees)"
profile_d_psi,s,h,"  2.0",,," Latitude binsize        of profiles (degrees)"
```

The following explains tutorial-style the meaning and reasons for the choice of these parameters. Note that the full explanation of each parameter is in the chapter ”Help File”.

```
################################################################################
#
# =#= spiskymax 20.0 =#=
# ~aws/integral/spectrometer/simulations/sources.3
```

These are comments, the user is free to modify them for documenting the run.

```
debug  ,i,h,0,0,2,"0=silent,1,2 gives more verbose output"
```

Output to log file (e.g. setenv COMMONLOGFILE mylog); however the diagnostic output will go to the screen unless piped to a file.

```
display,i,h,0,0,2,"1,2 displays input sky using root "
title  ,s,h,"3C273, 400-1000 keV, 10^6 sec  ",,,"title to write as comment to FITS header"
```

The display option is limited, useful for testing. The title is used to write to the FITS header of the output image file and hence useful for identifying the run.

```
rogroup,s,h,"",,,"xxx R/O Group"
rwgroup,s,h,"",,,"xxx R/W Group"
```

These DAL groups are not used in this run, but could be used as alternative to specifying the datasets as below.

```
counts_input_file,s,h,"SPI-OBS-DSP.spiskycnv_14.fits[1]",,,"xxx input count  file"
pointing_input_file,s,h,"SPI-OBS-PNT.spisimprep_2.fits[1]",,,"xxx input pointing  file"
ebounds_input_file, s,h,"SPI-EBDS-SET.spiskycnv_14.fits[1]",,,"xxx input energy bounds  file"
deadtime-dol,s,h,"SPI-OBS-DTI.spisimprep_2.fits[1]",,,"xxx DTI deadtime/livetime input file"
background_input_file,s,h,"SPI-BMOD-DSP-IDX.spiback.fits[1]",,,"xxx input background file"

gti-dol,s,h,"SPI-OBS-GTI.spisimprep_3.fits[1]",,,"GTI goodtime input file"
psd-efficiency-dol,s,h,"SPI-OBS-PEF.spisimprep_3.fits[SPI.-OBS.-PEF]",,,"input PSD efficiency file"
psd-response-dol,s,h,"SPI-OBS-PRF.spisimprep_3.fits[SPI.-OBS.-PRF]",,,"  input PSD response   file"
```

The input data as described in Chapter "Help File". The 'gti' and two 'psd' entries are required from Version 26.

```
irf_input_file,s,h,"${ISDC_TEST_DATA_DIR}/ic_tree/4.5/ic/spi/rsp/spi_irf_grp_0011.fits[GROUPING]" ,,," input IRF file 85 detectors"
```

This Instrument Response Function contains response for single, doubles and triples (detectors 0-84), and spiskymax uses the SPI-OBS-GTI (good times) dataset for the definition of the detectors present in the SPI-OBS-DSP detector spectra. This IRF is available at the ISDC under the "ic" (Instrument Characteristics) directory.

```
source-cat-dol,s,h,"SPI-SRCL-CAT.test_catalogue.fits[1]",,,"input catalogue of sources "
```

The catalogue of sources for which the flux is required. The user typically edits this file with fv to define the source positions, and sets the SEL_FLAG column to 1 for the chosen sources (the latter is convenient if the catalogue already exists and only a selection is being made).

```
image-idx,s,h , "SPI-SKY-IMA-IDX.spiskymax_19_back1.fits(SPI.-SKY.-IMA-IDX.tpl)",,,"output skymap image index file"
```

```
image-int,s,h ,     "SPI-SKY-IMA.spiskymax_19_back1.fits",,,"    output skymap images file"
```

The user is free to choose the name of the output image files, both for the index and the actual images and profiles. Note that while the index needs the template SPI.-SKY.-IMA-IDX.tpl, the actual images do not.

```
source-res-dol,s,h,"SPI-SRCL-RES.spiskymax_19_back1.fits(SPI.-SRCL-RES.tpl)",,,"output catalogue of sources "
```

The output catalogue containing the fluxes and errors of the sources selected in the input catalogue.

```
skymap_system,s,h,G,,,"input skymap coordinate system C=celestial, G=Galactic"
```

In SURVEY mode the user has the choice of specifying whether the skymap should be in the Celestial or Galactic system. The coordinates of the map (*chi_0* etc) are interpreted as belonging to the respective system.

```
image-fov,s,h,"SURVEY",,,"Image field of view: SURVEY  USER  POINTING POINTING+FCFOV POINTING+ZCFOV POINTING-CENTER"
```

SURVEY enables explicit definition of the field. The other options are useful for automatic processing and are included for consistency with *spiros*.

```
center-long,r,h, 1.0,,,"Longitude of centre pixel (degrees) [image-fov= USER]"
center-lat ,r,h, 2.0,,,"Latitude  of centre pixel (degrees) [image-fov= USER]"
image-pixel-long,r,h,0.5,,,"Longitude pixel size     (degrees) [image-fov= USER,POINTING*]"
image-pixel-lat,r,h,0.5,,,"Latitude  pixel size      (degrees) [image-fov= USER,POINTING*]"
image-dim-long,i,h,40,,,"Longitude dimension        (pixels) [image-fov= USER,POINTING-CENTER]"
image-dim-lat ,i,h,41,,,"Latitude  dimension        (pixels) [image-fov= USER,POINTING-CENTER]"
```

Since we have selected SURVEY, the image will be defined by the 6 parameters (*chi_0* etc), and *centre-long.....image_dim_lat* will be ignored.

```
chi_0, r,h, -20.0,,,"Longitude of first pixel (degrees)[image-fov= SURVEY]"
chi_1, r,h, +20.0,,,"Longitude of last  pixel (degrees)[image-fov= SURVEY]"
d_chi, r,h,   0.2,,,"Longitude binsize        (degrees)[image-fov= SURVEY]"
psi_0, r,h, -20.0,,,"Latitude  of first pixel (degrees)[image-fov= SURVEY]"
psi_1, r,h, +20.0,,,"Latitude  of last  pixel (degrees)[image-fov= SURVEY]"
d_psi, r,h,   0.2,,,"Latitude  binsize        (degrees)[image-fov= SURVEY]"
```

The boundaries of the image and the binning are defined here. Although labelled "Longitude" etc. the coordinates are interpreted according to *skymap_system*.

```
energy_range_min,i,h,  1,,,"minimum energy range sequence number as in ebounds file: 1,2,3..."
energy_range_max,i,h,  1,,,"maximum energy range sequence number as in ebounds file: 1,2,3..."
```

The input counts file may contain many energy ranges, and here we choose to analyse just the first one.

```
max_iter,i,h , 30,0,1000,"maximum number of memsys5 iterations"
image_default, r,h,1.0e-3,,,"image default intensity, cm-2 sr-1 s-1"
```

The maximum number of iterations should normally be set to a number like 30 for the first run, but it is normal to try different values up to 200 to see the effect on the image. The image default should not have much influence on the final image, but it should also be varied to test the effect. The value used here is found to be satisfactory.

```
background_scaling_default,r,h,1.0e-0,,,"background scaling default"
background_entropy_factor ,r,h,1.0e-0,,,"background factor to reduce contribution to entropy"
background_method,i,h,1,1,2,"background method:1=free 2=fixed"
```

The background scaling factor is here chosen to be 1 so that the entropy of the background will have a relatively large influence on the result. It is worth trying smaller values and in this case it is recommended to use

background_scaling_default=background_entropy_factor.

The background method is here chosen "free", i.e. the background is fitted based on the input template. If the background were accurately known from some other method then it might be appropriate to choose "fixed".

```
memsys_iBayes,i,h,1,1,3,"memsys Bayesian method code: 1= classic,  3=ad hoc specified by fAim    "
memsys_iEntropy,i,h,1,1,5,"memsys entropy type:         1= standard, 5=fixed total=sum over defaults"
memsys_fAim,r,h,1.0e-00,1e-10,1e10,"memsys stopping criterion: classic: 1/Omega =1.0, ad hoc: alpha "
```

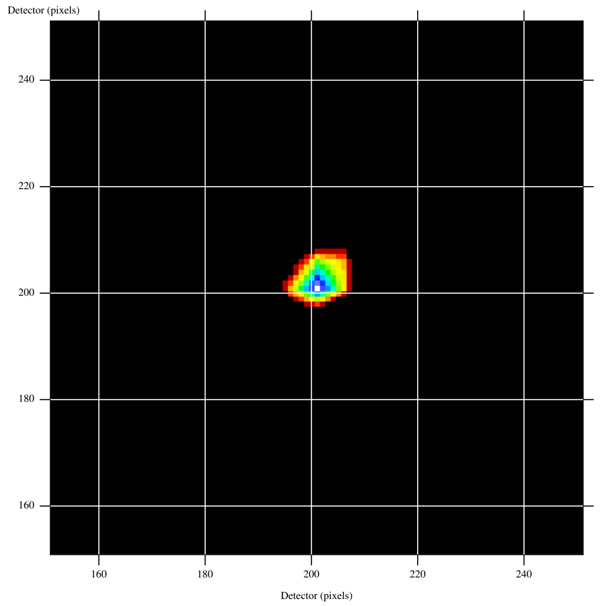
See the MEMSYS5 manual for explanations of these parameters. They can be safely left at these values in the normal case. If it is found that the automatic criterion stops too early, or if it is required to make a 'deeper' image, i.e. fitting the data better at the expense of more noise, then use *memsys_iBayes =3* and set *memsys_fAim = 1.e-10*; the iterations will then simply continue up to the maximum since the automatic stopping criterion will never be satisfied. In fact for *memsys_iBayes =3* , *memsys_fAim* defines the value of α (see 'MaxEnt Formulae') at which the iterations will stop, so reducing it forces the iterations to continue beyond the 'Classic MaxEnt' criterion.

```
n_source_fluxes,i,h,1,0,100,"number of source fluxes with errors"
source_chi_1,r,h,0.0,,,"longitude (degrees) of source 1"
source_psi_1,r,h,0.0,,,"latitude  (degrees) of source 1"
source_ON_radius_1, r,  h, 2.00 ,,,"ON radius (degrees)  for  source 1"
source_OFF_radius_1,r,h, 5.00,,,"OFF radius (degrees) for  source 1"
```

The sources are actually defined in the input source catalogue, but one source has to be specified here in order to define the ON and OFF radius. See the Chaper "Help File" for details of the method and meaning of these parameters. Since the catalogue is used, *source_chi_1,source_psi_1* are actually ignored.

```
n_profiles,i,h,0,0,100,"number of intensity profiles with errors"
profile_chi_0,s,h,"-20.0",,,"Longitude of first pixel of profiles (degrees)"
profile_chi_1,s,h,"+20.0",,,"Longitude of  last pixel of profiles (degrees)"
profile_d_chi,s,h,"  2.0",,,"Longitude binsize        of profiles (degrees)"
profile_psi_0,s,h," -0.0",,," Latitude of first pixel of profiles (degrees)"
profile_psi_1,s,h," +0.0",,," Latitude of  last pixel of profiles (degrees)"
profile_d_psi,s,h,"  2.0",,," Latitude binsize        of profiles (degrees)"
```

No profiles are requested in this case so these parameters are ignored. For a case with profiles see '511 keV diffuse emission'.



The image shows the source clearly at the correct position (centred). Note the pixel size is 0.2^o.

4.2 511 keV diffuse emission

This example is a simulation of the GCDE observation of the diffuse 511 keV line based on OSSE results. 2 GCDE cycles of 4.8 10^6 seconds each are used. Two profiles, in longitude and latitude, are generated, illustrating the usefulness of the error-bar generation function.

The spiskymax.par file is:

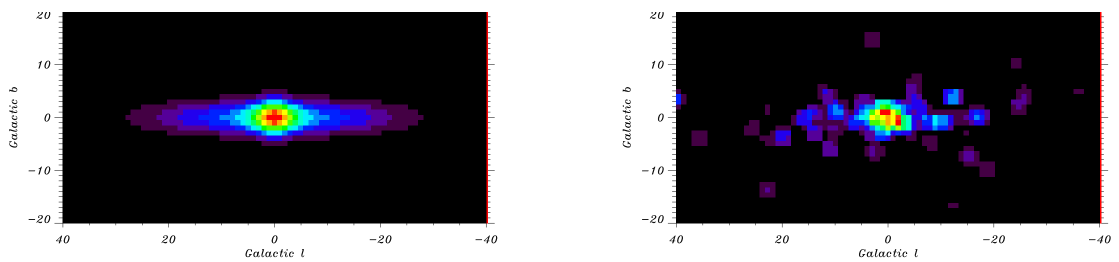
```
################################################################################
#
# =#= spiskymax 20.0 =#=
# ~aws/integral/spectrometer/simulations/gcde.18

debug  ,i,h,1,0,2,"0=silent,1,2 gives more verbose output"
```

```
display,i,h,0,0,2,"1,2 displays input sky using root "
title  ,s,h,"GCDE 511 keV ",,,"title to write as comment to FITS header"
rogroup,s,h,"",,,"xxx R/O Group"
rwgroup,s,h,"",,,"xxx R/W Group"
counts_input_file,s,h,"SPI-OBS-DSP.spiskycnv_14.fits[1]",,,"xxx input count  file"
pointing_input_file,s,h,"SPI-OBS-PNT.spisimprep_2.fits[1]",,,"xxx input pointing  file"
ebounds_input_file, s,h,"SPI-EBDS-SET.spiskycnv_14.fits[1]",,,"xxx input energy bounds  file"
deadtime-dol,s,h,"SPI-OBS-DTI.spisimprep_2.fits[1]",,,"xxx DTI deadtime/livetime input file"
background_input_file,s,h,"SPI-BMOD-DSP-IDX.spiback.fits[1]",,,"xxx input background file"

irf_input_file,s,h,"/afs/ipp/mpe/gamma/integral/isdc/data/spi/IRFs/spi_irf_grp_0005.fits[GROUPING]" ,,," input IRF file 19 detectors"


source-cat-dol,s,h,"SPI-SRCL-CAT.test_catalogue.fits[1]",,,"input catalogue of sources "
image-idx,s,h ,"SPI-SKY-IMA-IDX.spiskymax_20.fits(SPI.-SKY.-IMA-IDX.tpl)",,,"output skymap image index file"
image-int,s,h ,"SPI-SKY-IMA.spiskymax_20.fits",,,"xxx Output skymap images file"

source-res-dol,s,h,"SPI-SRCL-RES.spiskymax_20.fits(SPI.-SRCL-RES.tpl)",,,"output catalogue of sources "
skymap_system,s,h,G,,,"input skymap coordinate system C=celestial, G=Galactic"
image-fov,s,h,"SURVEY",,,"Image field of view: SURVEY  USER  POINTING POINTING+FCFOV POINTING+ZCFOV POINTING-CENTER"
center-long,r,h, 1.0,,,"Longitude of centre pixel (degrees) [image-fov= USER]"
center-lat ,r,h, 2.0,,,"Latitude  of centre pixel (degrees) [image-fov= USER]"
image-pixel-long,r,h,0.5,,,"Longitude pixel size     (degrees) [image-fov= USER,POINTING*]"
image-pixel-lat,r,h,0.5,,,"Latitude  pixel size      (degrees) [image-fov= USER,POINTING*]"
image-dim-long,i,h,40,,,"Longitude dimension        (pixels) [image-fov= USER,POINTING-CENTER]"
image-dim-lat ,i,h,41,,,"Latitude  dimension        (pixels) [image-fov= USER,POINTING-CENTER]"

chi_0, r,h, -40.0,,,"Longitude of first pixel (degrees)[image-fov= SURVEY]"
chi_1, r,h, +40.0,,,"Longitude of last  pixel (degrees)[image-fov= SURVEY]"
d_chi, r,h,   1.0,,,"Longitude binsize        (degrees)[image-fov= SURVEY]"
psi_0, r,h, -20.0,,,"Latitude  of first pixel (degrees)[image-fov= SURVEY]"
psi_1, r,h, +20.0,,,"Latitude  of last  pixel (degrees)[image-fov= SURVEY]"
d_psi, r,h,   1.0,,,"Latitude  binsize        (degrees)[image-fov= SURVEY]"

energy_range_min,i,h,  1,,,"minimum energy range sequence number as in ebounds file: 1,2,3..."
energy_range_max,i,h,  1,,,"maximum energy range sequence number as in ebounds file: 1,2,3..."

max_iter,i,h , 30,0,1000,"maximum number of memsys5 iterations"
image_default, r,h,1.0e-3,,,"image default intensity, cm-2 sr-1 s-1"
background_scaling_default,r,h,1.0e-0,,,"background scaling default"
```

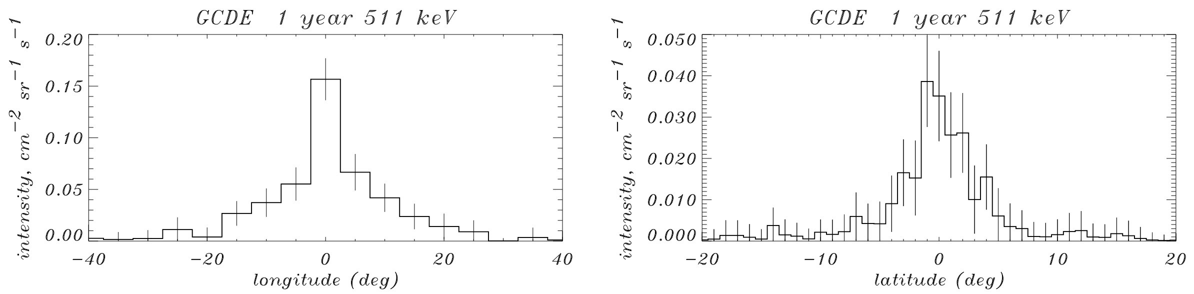
```
background_entropy_factor ,r,h,1.0e-0,,,"background factor to reduce contribution to entropy"
background_method,i,h,2,1,2,"background method:1=free 2=fixed"

memsys_iBayes,i,h,1,1,3,"memsys Bayesian method code: 1= classic,  3=ad hoc specified by fAim    "
memsys_iEntropy,i,h,1,1,5,"memsys entropy type:         1= standard, 5=fixed total=sum over defaults"
memsys_fAim,r,h,1.0,1e-10,1e10,"memsys stopping criterion: classic: 1/Omega =1.0, ad hoc: alpha "

n_source_fluxes,i,h,1,0,100,"number of source fluxes with errors"
source_chi_1,r,h,0.0,,,"longitude (degrees) of source 1"
source_psi_1,r,h,0.0,,,"latitude  (degrees) of source 1"
source_ON_radius_1, r,  h, 0.50 ,,,"ON radius (degrees)  for  source 1"
source_OFF_radius_1,r,h, 2.00,,,"OFF radius (degrees) for  source 1"


n_profiles,i,h,2,0,100,"number of intensity profiles with errors"
profile_chi_0,s,h,"-40.0  0.0",,,"Longitude of first pixel of profiles (degrees)"
profile_chi_1,s,h,"+40.0  0.0",,,"Longitude of  last pixel of profiles (degrees)"
profile_d_chi,s,h,"  5.0 80.0",,,"Longitude binsize        of profiles (degrees)"
profile_psi_0,s,h," -0.0 -20.0",,," Latitude of first pixel of profiles (degrees)"
profile_psi_1,s,h," +0.0 +20.0",,," Latitude of  last pixel of profiles (degrees)"
profile_d_psi,s,h,"  5.0   1.0",,," Latitude binsize        of profiles (degrees)"
```

The only part of this parameter file which is not covered in the '3C273' example concerns the profiles. Here we have chosen 2 profiles, one in longitude and one in latitude. Each parameter entry is a vector giving the values for each profile, e.g. *profile_chi_0,s,h,"-40.0 0.0"* means that the first profile has a value -40.0, the second profile has a value 0.0 for this parameter. The longitude profile extends from l=-40^o to +40^o with a binsize of 5^o in longitude and latitude, centred on b=0. The latitude profile extends from b=-20^o to +20^o with a binsize of 80^o in longitude (i.e. integrated over the full longitude range of the map) and 1^o in latitude.

Note that profiles do not have to be one-dimensional as here; the specification above allows 2D profiles (i.e. rebinned images) to be defined and they will be correctly computed and output.



The input image (left) and the *spiskymax* image (right).



Longitude and latitude profiles.

Chapter 5

FAQ, Tips and tricks

This section is intended to collect questions and answers, tips and experiences arising from use of *spiskymax* .

1. Q: how many pointings are necessary to get a good image ?

   A: if the background is known then one pointing and background_method = 2 (fixed) will generate some kind of image, but several pointings improves things a lot. If the background is unknown then more than one pointing is absolutely necessary since it has to be fitted with background_method = 1 (free) and there is not enough information to do this from a single pointing.

2. Q: What value for 'maximum iterations' ?

   A: Try 30 and look at the diagnostics. If the iterations stop automatically before this number then it is sufficient. If not try more. Examine the image for signs of 'over-fitting' (clumpiness) and artefacts. For sources fluxes, increase the iterations until a constant value is reached. Experiment with the image and background default values.

3. Q: is *spiskymax* good for analysis of sources?

   A: Sources are detected as well as in other methods, in particular multiple sources are no problem. For fluxes, if the position is known then there is no reason for the results to be inferior to other methods. One disadvantage is that the 'ON' and 'OFF' regions have to be defined.

4. Q: what if the CPU time is very long?

   A: this can occur if the pixel-size is small and the map is large. Try increasing the pixel-size. Generally start with 1^o pixels and when the map is satisfactory reduce the size to 0.5^o or 0.1^o.

5. Q: are pseudo-detectors (i.e. multiple events) an advantage over single events?

   A: definitely. They increase the data available especially at higher energies, and have lower background than singles. They do increase the CPU time but this is not normally a problem.

6. Q: what does an error message from 'common_prepare_pars' indicate ?

   A: often it means that some input data cannot be found, or that the output data already exists (always delete old output files before running - applies to any ISDC executable).

7. Q: who to ask for more information?

   A: either the ISDC help desk or the developer of *spiskymax* : aws@mpe.mpg.de

Chapter 6

Simulating observations

Simulation is an important part of understanding data analysis, both for testing methods, planning observations and their analysis, and understanding the actual results. Although not part of *spiskymax* , a description of how to do simulations is given here since anyone using *spiskymax* (or other analysis tools) will often need it. The procedure differs from the usual one for analysing real data due to the need to create all the relevant data files 'from scratch'. The steps are:

1. Generate a pointing file (ATTI) using *os_pdefgen*
2. Generate a model intensity skymap (SKY-IMA) using *gensky*
3. Generate a good time intervals, dead times, good times, PSD efficiency, PSD response, pointing and energy bounds (GTI,DTI,GTI,PEF,PRF,PNT,EBDS) files using *spisimprep*
4. Generate a background template (BMOD)using *spiback*
5. Generate simulated count spectra (DSP) using *spiskycnv*
6. Run *spiskymax* (and/or *spiros, spidiffit*)

   NB The generation of GTI and PSD data (required by spiskymax 26.0) is new in spisimprep 3.0, spiskycnv 16.0.

os_pdefgen is part of the ISDC Observation Simulator and allows a fairly flexible definition of pointing sequences, including raster scans of any dimension and pitch. *gensky* allows skymaps to be defined as a function of energy. It builds the map out of any number of user-definable components, which include point-sources, extended Gaussian sources, input maps like HI, CO surveys, as well as constant, line and power-law spectra. *spisimprep* is a utility which is required just to satisfy some interfaces needed by following programs like *spiback*. *spiback* generates the background template. *spiskycnv* convolves the input map with the instrumental response for each of the defined pointings with their observation times (allowing for deadtime) and generates Poisson simulated count spectra. This completes the data preparation, which is now ready for *spiskymax* analysis.

The details of how to run each of these executables are given in the respective help files (executable_name.txt) which can be viewed by typing eg 'os_pdefgen -h'

Provided all the file names are consistently defined, and the parameter files are present in directory given by $PFILES, the following one-line command will perform the entire simulation + analysis:

```
os_pdefgen; gensky; spisimprep; spiback; spiskycnv; spiskymax
```

Chapter 7

Help File

This is a copy of the 'help file' spiskymax.txt, which is maintained in detail for each version of *spiskymax* and should be consulted for precise definitions of parameters and usage, and for version-specific changes.

```
NAME
spiskymax v26.0

USAGE
spiskymax

NB For simulations, this executable depends on previous runs
of  gensky, os_pdefgen,spipoint, spidead, spiback and
spiskycnv.  (More conveniently  simulations can be performed
with spisimprep instead of spipoint and spidead especially for
large survey applications.)

For real data, depends on previous runs of spipoint, spidead, spigti,
spi_psd_effigen, spi_psd_respgen, spiback and spihist.

        Manual: The spiskymax Users Manual, including description of the method
        and parameters, with examples, is included in this
        delivery (spiskymax_manual_v3.ps).
        The current version of the manual can be
        found at http://www.mpe.mpg.de/~aws/integral.html


        Algorithm:  Users interested in a complete explanation of the
```

```
principles of the method should consult the MEMSYS5 User's
Manual (to be found at http://www.maxent.co.uk)


        Platform: the makeisdc1.in cover both Solaris and Linux cases
        ** using isdc_config and makefiles-2.4.9 **. This was *new* in spiskymax v24,
        (previous version required manual editing of makeisdc1.in)

        The ISDC GUI tool param_gui can be used;
        to run the GUI:   "param_gui spiskymax.par"

DESCRIPTION

       Produces an intensity skymap from count data by deconvolving
with the SPI instrument response using the Maximum Entropy
method.  Analyses sources at specified positions.

The instrument response is in the form of IRF (Instrument
Response Function) data as provided by GSFC using GEANT
simulations. From spiskymax Version 26.0 on this is flexible:
        the detector IDs are taken from the SPI.-OBS.GTI dataset.
        Hence use an IRF for detectors 0-84 which covers all cases
        (singles, doubles, triples).

Creates skymaps in form of  SPI.-SKY.-IMA-IDX (index)
 and SPI.-SKY.-IMA (images) dataset.

       The pointings are defined by a SPI.-OBS.-PNT dataset.

       Deadtime is defined by a SPI.-OBS.-DTI dataset.

       Detector ID definitions are taken from a SPI.-OBS.GTI dataset.

       PSD efficiency and response are defined by
        SPI.-OBS.-PEF and SPI.-OBS.-PRF datasets
        These are used for detectors 85-141 which use PSD information.

       Background is defined by the output of spiback, in the form of
SPI.-BMOD-DSP-IDX (index) and SPI.-BMOD-DSP (data). The
background values for each detector are determined, relative to
```

```
the input values, as part of the reconstruction process ,if
back_method=1. They may also be fixed at the values in the
input background dataset (background_method=2); this is
appropriate if the background was determined by some
INDEPENDENT method such as, for a line, interpolating between
adjacent (non-line) ranges.

The input count data  may contain multiple energies, in which
case the deconvolution is performed for a range of energies
which can be specified.

       Source fluxes:
After the image is produced, the flux and its error for a
number of user-defined sources can be optionally determined.
Each source is specified in terms of a position and  'ON' and
'OFF' radii. The ON region is the circle centred on the source
position with radius ON and the OFF region is the annulus
between the ON and OFF radii.  The source flux is defined as

 sum(intensity in ON region)/sum(intensity in OFF region)
                                    *solid angle(ON)/solid angle(OFF).

The error is based on a Bayesian analysis which marginalizes
over all the unwanted degrees of freedom, so the significance
of a source by this method may not correspond to a 'classical'
significance level. In general significances increase for
smaller skymaps since the degrees of freedom decrease.

Source fluxes can be generated based either on the list in the
parameter file (with output only to LOG_1) or based on an input
catalogue specified in the parameter file. The catalogue method
is better since the output is stored and existing catalogues
can be used. An output catalogue is produced containing the
source fluxes and errors (columns FLUX and FLUX_ERR) for all
energies specified in the SPI-EBDS-SET file. The input
catalogue must be in the standard ISDC format defined in
SPI.-SRCL-CAT.tpl and described in the DAL3CAT library
documentation (see ISDC WWW site). The output catalogue uses
SPI.-SRCL-RES.tpl (which is essentially the same for this purpose as
GNRL-REFR-CAT). The only information used  from the input
```

```
catalogue is the sources' RA and dec; the rest is simply copied
to the output catalogue. Only sources with SEL_FLAG=1 in the
input catalogue are selected for analysis and placed in the
output catalogue.  The ON and OFF radii for all catalogue
sources are taken from source_ON_radius_1, source_OFF_radius_1
in the parameter file, so that at least one source must also be
specified in the parameter file (can have a dummy position).  The
ON and OFF radii are recorded in the COMMENTS column of the
output catalogue.

       Profiles:
After the image is produced, profiles of intensity may be
generated from a number of user-defined specifications. A
'profile' is an image with both intensity and error on
intensity.  Normally it is one-dimensional but  may be
generalized to a 2D map for which each pixel has intensity and
error. Each profile is defined by the start and end points in
both dimensions and the pixel size in both dimensions. The
profiles are output to the same  SPI.-SKY.-IMA dataset as the
images, and are indexed in SPI.-SKY.-IMA-IDX. For each profile
the intensity and error occupy separate extensions with IMATYP
='intensity' and 'error' as foreseen for this data type.


CHANGES
Version 12 Output skymap image index parameters changed to image-idx
           Output skymap image index template changed to
            SPI.-SKY.-IMA-IDX for compatibility with ISDC v2.1
Version 13 ISDC v2.1 with Templates 2.1.3 : compatible with gensky 6.0 spiskycnv 11.0.
           however still using old SPI.-BMOD-IDX.tpl ie. compatible with spiback 1.2
Version 14 Fully compatible with templates 2.1.3 including SPI.-BMOD-DSP-IDX.tpl
           Hence use only with spiback >1.2
Version 15 changes in makeisdc1.in for root v3.00.06. Works with isdc v2.2
Version 16 improved  makeisdc1.in for more portability and Sun Workshop 6.1
Version 17 analysis of sources in catalogue - NB requires dal3cat Version >=3.0 to correctly write the output catalogue; improved colour bar for graphics output. LINUX compatibility changes, but makeisdc1.in does not link for LINUX
Version 18 General code readability improvements and compatibility with spiskycnv v14. Link with spibham v1.3.1 (v2.1 not yet working).
Version 19. Catalogue input SPI.-SRCL-CAT instead of GNRL-REFR-CAT.
All output via RIL, except for output from MEMSYS5 package.
Version 20. Corrected handling of background_method=2: fixed background
Version 21. Corrected FITS head of first image when profiles requested (SPR-1161)
```

```
Version 22. makeisdc1.in includes linux option
            SPR1169 fixed (blanks in spiskymax.par)
            SPR1171 fixed (dal_element=NULL)
            Use $ISDC_TEST_DATA_DIR for IRFs in spiskymax.par
            NB requires dal3cat v3.1.1 (3.0.1 does not work correctly)

Version 23  spiskymax.par compatible with isdcroot (SPR-1221)
            write_SKY_IMA.cpp, write_profiles.cpp: remove multiple timestamps (SPR-1235)


Version 24  avoid zero-divide problems when source flux error =0 (SPR-1604)
            makeisdc1.in uses isdc-config for automatic f90 support under Solaris/Linux
            NB needs makefiles 2.4.9 and $ISDC_ENV/bin in $PATH
            GUI: isdc param_gui can be used on spiskymax_gui.par as delivered

Version 25  SPR-1691: Y_axis.cpp problem solved. SPR-1238 ITE_NUM keyword written.

Version 26. SCREW-779 flexible input IRF: detector IDs from SPI-OBS-GTI.
            PSD response handled using SPI-OBS-PEF and and SPI-OBS-PRF datasets.

===============================================================================================


PARAMETERS

        debug
         enables debugging. Normally 0.
         0=silent. 1,2 gives more verbose output

        display
         enables display of skymaps. Normally 0.
         1,2 displays output skymaps using root. 1=image format, 2=image and lego formats.
         The displays can be output to postscript and other formats via the interactive canvas
         which is presented.

        title
         user's description of the run: serves 2 purposes
         1. identifies this run (via the par file)
         2. used to write as comment to FITS header of the output skymap datasets.
```

```
rogroup
    read only group for input. If empty, uses the filenames directly.

rwgroup
write group for output.  If empty, no group is created.

        counts_input_file
         input detector count spectra file (template SPI.-OBS.-DSP)

        pointing_input_file
         input pointing file               (template SPI.-OBS.-PNT)

        ebounds_input_file
         input energy bounds file          (template SPI.-EBDS-SET)

        deadtime-dol
          input dead time file             (template SPI.-OBS.-DTI) created by spidead.

        gti-dol
          input good time file             (template SPI.-OBS.-GTI) created by spigti.

        irf_input_file
          input IRF index file which contains the SPI response information
          as generated by GEANT simulations at GSFC. The current IRF including
          singles+doubles+triples is  spi_irf_grp_0011.fits
          These IRFs are in the ic (Instrument Characteristics) area at ISDC.
          At present this can be found at (under standard ISDC environment)
          ${ISDC_TEST_DATA_DIR}/ic_tree/4.5/ic/spi/rsp



        psd-efficiency-dol
          input PSD efficiency file        (template SPI.-OBS.-PEF) created by spi_psd_effigen.

        psd-response-dol
          input PSD response file          (template SPI.-OBS.-PRF) created by spi_psd_respgen.

        background_input_file
         input background index file       (template SPI.-BMOD-IDX)
```

```
source-cat-dol
  input catalogue dataset (template SPI.-SRCL-CAT)
  This specifies the source positions to be analysed. Select sources using SEL_FLAG.
  [A source list in this format can be generated from GNRL-REFR-CAT (using ISDC tools ?)
  or output from spiros can be used.]


image-idx
 output skymap index file (template SPI.-SKY.-IMA-IDX.tpl)

image-int
 output skymap images file  (template SPI.-SKY.-IMA.tpl)

source-res-dol
 output source catalogue containing flux spectra with error bars (template SPI.-SRCL-RES.tpl)

 skymap_system
  input skymap coordinate system C=celestial, G=Galactic"
  This tells spiskycnv how to treat the input skymap, which does not
  itself specify the basis system. Usually Galactic is used.

 chi_0
 Longitude of first pixel (degrees). Centre of pixel (as FITS convention)
 chi_1
 Longitude of last pixel (degrees).  Centre of pixel (as FITS convention)
 d_chi
  Longitude binsize (degrees).

 psi_0
  Latitude of first pixel (degrees). Centre of pixel (as FITS convention)

 psi_1
  Latitude of last pixel (degrees). Centre of pixel (as FITS convention)

 d_psi
  Latitude binsize (degrees).

 energy_range_min
  minimum energy range sequence number as in ebounds file: 1,2,3...
  energy ranges energy_range_min to energy_range_max are processed.
```

```
 energy_range_max
  maximum energy range sequence number as in ebounds file: 1,2,3...



image-fov
 Choice of how the image field of view is defined.
 Must be one of: SURVEY  USER  POINTING  POINTING+FCFOV  POINTING+ZCFOV  POINTING-CENTER

 SURVEY: maps limits and binsize using chi_0,chi_1,d_chi,psi_0,psi_1,d_psi
 USER  : map  centre and dimensions using center-long center-lat etc as below
 POINTING      : use extremes of pointing directions
 POINTING+FCFOV: use extremes of pointing directions + fully coded FOV ( 7 degrees)
 POINTING+ZCFOV: use extremes of pointing directions + zero  coded FOV (17 degrees)
 POINTING-CENTER: use average pointing and dimension defined by image-dim-long,image-dim-lat

 Apart from SURVEY the pixel size is defined by image-pixel-long,image-pixel-lat.
 USER  POINTING POINTING+FCFOV POINTING+ZCFOV POINTING-CENTER are spiros-compatible definitions.

center-long
  Longitude of centre pixel (degrees) [image-fov= USER]

center-lat
  Latitude  of centre pixel (degrees) [image-fov= USER]

image-pixel-long
  Longitude pixel size      (degrees) [image-fov= USER,POINTING*]

image-pixel-lat
  Latitude  pixel size      (degrees) [image-fov= USER,POINTING*]

image-dim-long
  Longitude dimension        (pixels) [image-fov= USER,POINTING-CENTER]

image-dim-lat
  Latitude  dimension        (pixels) [image-fov= USER,POINTING-CENTER]
```

```
       max_iter
maximum number of memsys5 iterations Normally the program
should finish automatically (using a stop criterion) within 50
iterations. If not this parameter can be increased. If the
limit is reached without the stop criterions being attained,
the skymap is normally still a good approximation.


       image_default
image default intensity, cm-2 sr-1 s-1. The value to be
assigned to pixels which are unconstrained by the data. Set to
the expected average intensity.  This parameter can be adjusted
to optimize the maps but the result should not be too sensitive
to its value.

       background_scaling_default
background scaling default value, normally 1.0. Required since
the background is determined in the imaging process by the same
process as the skymap.  A value 1.0 indicates that the input
background model is default.

       background_entropy_factor
background factor to reduce contribution to entropy. Normally
1.0 Since image and background are treated in the same way from
the viewpoint of entropy while the background should strictly
not contribute to the entropy, this scaling can be used to
reduce this contribution to a negligible level

       background_method
Determines whether the background is to be fitted ('free') or
fixed at the level of the input background.  background
method:1=free 2=fixed

      memsys_iBayes
       memsys Bayesian method code as defined in the MEMSYS5 User's
       Manual (see SEE ALSO) 1= classic,  3=ad hoc specified by fAim
```

```
memsys_iEntropy
 memsys entropy type  as defined in the MEMSYS5 User's Manual
 1= standard, 5=fixed total=sum over defaults

memsys_fAim
 memsys stopping criterion  as defined in the MEMSYS5 User's Manual
 classic: 1/Omega =1.0, ad hoc: alpha "

n_source_fluxes
 number of source fluxes with errors to be computed
 The sources are specified in the following parameters:

source_chi_1,
 longitude (degrees) of source 1
 See the DESCRIPTION section for explanation of this and following parameters.

source_psi_1
 latitude  (degrees) of source 1

source_ON_radius_1
 ON radius (degrees)  for  source 1

source_OFF_radius_1,
 OFF radius (degrees) for  source 1

source_chi_2
longitude (degrees) of source 2

etc. for all sources


n_profiles
 number of intensity profiles with errors

profile_chi_0
 Longitude of first pixel of profiles (degrees); vector of values
 as string separated by spaces

profile_chi_1
 Longitude of  last pixel of profiles (degrees); vector of values
```

```
 as string separated by spaces

profile_d_chi
 Longitude binsize        of profiles (degrees); vector of values
 as string separated by spaces

profile_psi_0
 Latitude of first pixel  of profiles (degrees); vector of values
 as string separated by spaces

profile_psi_1
 Latitude of  last pixel of profiles (degrees) ; vector of values
 as string separated by spaces

profile_d_psi
 Latitude binsize        of profiles (degrees) ; vector of values
 as string separated by spaces
```

EXAMPLE

```
################################################################################
#
# =#= spiskymax 26.0 =#=
#
################################################################################
debug  ,i,h,0,0,2,"0=silent,1,2 gives more verbose output"
display,i,h,0,0,2,"1,2 displays input sky using root "
title  ,s,h,"Version 26.0 unit test Detectors 19-84, 123-141 ",,,"title to write as comment to FITS header"
rogroup,s,h,"",,,"xxx R/O Group"
rwgroup,s,h,"",,,"xxx R/W Group"
counts_input_file,s,h,"test_data/SPI-OBS-DSP.spiskycnv_16.fits[1]",,,"xxx input count  file"
pointing_input_file,s,h,"test_data/SPI-OBS-PNT.spisimprep_3.fits[1]",,,"xxx input pointing  file"
ebounds_input_file, s,h,"test_data/SPI-EBDS-SET.spiskycnv_16.fits[1]",,,"xxx input energy bounds  file"
deadtime-dol,s,h,"test_data/SPI-OBS-DTI.spisimprep_3.fits[1]",,,"xxx DTI deadtime/livetime input file"

gti-dol,s,h,"test_data/SPI-OBS-GTI.spisimprep_3.fits[1]",,,"GTI goodtime input file"
```

```
#irf_input_file,s,h,"${ISDC_TEST_DATA_DIR}/ic_tree/3.2/ic/spi/rsp/spi_irf_grp_0008.fits[GROUPING]",,,"input IRF file 19-84"
irf_input_file,s,h,"${ISDC_TEST_DATA_DIR}/ic_tree/4.5/ic/spi/rsp/spi_irf_grp_0011.fits[GROUPING]",,,"input IRF file"

psd-efficiency-dol,s,h,"test_data/SPI-OBS-PEF.spisimprep_3.fits[SPI.-OBS.-PEF]",,,"input PSD efficiency file"
psd-response-dol,s,h,"test_data/SPI-OBS-PRF.spisimprep_3.fits[SPI.-OBS.-PRF]",,,"  input PSD response   file"




background_input_file,s,h,"test_data/SPI-BMOD-DSP-IDX.spiback.fits[1]",,,"xxx input background file"



source-cat-dol,s,h,"test_data/SPI-SRCL-CAT.test_catalogue.fits[1]",,,"input catalogue of sources "
image-idx,s,h ,"out/SPI-SKY-IMA-IDX.spiskymax_26.fits(SPI.-SKY.-IMA-IDX.tpl)",,,"output skymap image index file"
image-int,s,h ,"SPI-SKY-IMA.spiskymax_26.fits",,,"xxx Output skymap images file"

source-res-dol,s,h,"out/SPI-SRCL-RES.spiskymax_26.fits(SPI.-SRCL-RES.tpl)",,,"output catalogue of sources "
skymap_system,s,h,C,,,"input skymap coordinate system C=celestial, G=Galactic"
image-fov,s,h,"SURVEY",,,"Image field of view: SURVEY  USER  POINTING POINTING+FCFOV POINTING+ZCFOV POINTING-CENTER"
center-long,r,h, 1.0,,,"Longitude of centre pixel (degrees) [image-fov= USER]"
center-lat ,r,h, 2.0,,,"Latitude  of centre pixel (degrees) [image-fov= USER]"
image-pixel-long,r,h,0.5,,,"Longitude pixel size     (degrees) [image-fov= USER,POINTING*]"
image-pixel-lat,r,h,0.5,,,"Latitude  pixel size      (degrees) [image-fov= USER,POINTING*]"
image-dim-long,i,h,40,,,"Longitude dimension        (pixels) [image-fov= USER,POINTING-CENTER]"
image-dim-lat ,i,h,41,,,"Latitude  dimension        (pixels) [image-fov= USER,POINTING-CENTER]"

chi_0, r,h, -20.0,,,"Longitude of first pixel (degrees)[image-fov= SURVEY]"
chi_1, r,h, +20.0,,,"Longitude of last  pixel (degrees)[image-fov= SURVEY]"
d_chi, r,h,   1.0,,,"Longitude binsize        (degrees)[image-fov= SURVEY]"
psi_0, r,h, -20.0,,,"Latitude  of first pixel (degrees)[image-fov= SURVEY]"
psi_1, r,h, +20.0,,,"Latitude  of last  pixel (degrees)[image-fov= SURVEY]"
d_psi, r,h,   1.0,,,"Latitude  binsize        (degrees)[image-fov= SURVEY]"

energy_range_min,i,h,  1,,,"minimum energy range sequence number as in ebounds file: 1,2,3..."
energy_range_max,i,h,  1,,,"maximum energy range sequence number as in ebounds file: 1,2,3..."

max_iter,i,h , 30,0,1000,"maximum number of memsys5 iterations"
```

```
image_default, r,h,1.0e-3,,,"image default intensity, cm-2 sr-1 s-1"
background_scaling_default,r,h,1.0e-0,,,"background scaling default"
background_entropy_factor ,r,h,1.0e-0,,,"background factor to reduce contribution to entropy"
background_method,i,h,2,1,2,"background method:1=free 2=fixed"

memsys_iBayes,i,h,1,1,3,"memsys Bayesian method code: 1= classic,  3=ad hoc specified by fAim    "
memsys_iEntropy,i,h,1,1,5,"memsys entropy type:         1= standard, 5=fixed total=sum over defaults"
memsys_fAim,r,h,1.0,1e-10,1e10,"memsys stopping criterion: classic: 1/Omega =1.0, ad hoc: alpha "

n_source_fluxes,i,h,1,0,100,"number of source fluxes with errors"
source_chi_1,r,h,0.0,,,"longitude (degrees) of source 1"
source_psi_1,r,h,0.0,,,"latitude  (degrees) of source 1"
source_ON_radius_1, r,  h, 0.50 ,,,"ON radius (degrees)  for  source 1"
source_OFF_radius_1,r,h, 2.00,,,"OFF radius (degrees) for  source 1"

#n_profiles,i,h,1,0,200,"number of intensity profiles with errors"
#profile_chi_0,s,h,"-20.0 -20.0 -20.0  +1.0  +1.0   0.0",,,"Longitude of first pixel of profiles (degrees)"
#profile_chi_1,s,h,"+20.0 +20.0 +20.0  +1.0  +1.0   6.0",,,"Longitude of  last pixel of profiles (degrees)"
#profile_d_chi,s,h,"  1.0   2.0   5.0   4.0   2.0   2.0",,,"Longitude binsize        of profiles (degrees)"
#profile_psi_0,s,h," -0.0   0.0  -0.0 -20.0 -20.0 -20.0",,," Latitude of first pixel of profiles (degrees)"
#profile_psi_1,s,h," +0.0  +0.0  +0.0 +20.0 +20.0 +20.0",,," Latitude of  last pixel of profiles (degrees)"
#profile_d_psi,s,h,"  2.0   2.0   3.0   1.0   2.0   1.0",,," Latitude binsize        of profiles (degrees)"

n_profiles,i,h,0,0,100,"number of intensity profiles with errors"
profile_chi_0,s,h,"-20.0",,,"Longitude of first pixel of profiles (degrees)"
profile_chi_1,s,h,"+20.0",,,"Longitude of  last pixel of profiles (degrees)"
profile_d_chi,s,h,"  2.0",,,"Longitude binsize        of profiles (degrees)"
profile_psi_0,s,h," -0.0",,," Latitude of first pixel of profiles (degrees)"
profile_psi_1,s,h," +0.0",,," Latitude of  last pixel of profiles (degrees)"
profile_d_psi,s,h,"  2.0",,," Latitude binsize        of profiles (degrees)"




BUGS
- Index of images is not written correctly (write_SKY_IMA:DAL error -1207) in
  some cases if n_profiles>0 and a subset of energies is requested.
  When all energies are processed there is no problem. Origin of error unclear,
  but it is not critical.
```

```
  Compilation of entire code with "CC -fast " leads to DAL error -1207 as above
  and failure to write keyworks to output SPI-OBS-IMA. Origin is unclear.
  However only the routines vopus.cpp, vtropus.cpp, memsys5.cpp and vector.cpp need
  to be optimized (and indeed should be for faster performance) and these cause no problem.

Send all bugs to aws@mpe.mpg.de

AUTHOR

A.W.Strong, MPE, 18 Sept  2002

aws@mpe.mpg.de

SEE ALSO
        SPI ICD
gensky.txt
        spipoint.txt
        spihist.txt
        spiback.txt
        spiskymax.txt
        MEMSYS5 User's Manual (to be found at http://www.maxent.co.uk)
        spiskymax User's Manual (http://www.mpe.mpg.de/~aws/integral.html)
```