The Error Propagation of the SWS Pipeline

Prepared by: E. Wieprecht, H.Feuchtgruber

1 Introduction

This paper sumarizes the error propagation of the SWS PIPELINE. It is based on the SWS PIPELINE version 8.4 which includes an improved error handling in the DERIVE AAR part and Auto Analysis product ([Feuchtgruber 1998]). The new approach separates pure statistical errors from systematic errors. It is possible to get a first indication of the accuracy of an SWS product from the product itself rather from documents only.

Error handling in Calibration Procedures (CAPs) which are used to derive calibration files and their error values are not discussed in this note. This is subject of CAP related documentation (e.g. IA help system).

Of course, some sources of error (e.g. fringes) will always depend on what sort of and how careful, data reduction has been done outside the automatic PIPELINE.

2 Statistical Errors

Within the PIPELINE chain there are not many sources providing information about statistical errors. The main statistical error is contained in the flag of the individual slope. NODATA-, GLITCH-, OOL-flags and the TINT tag (which gives the actual number of samples used to compute a ramp) provide rather reliable information about the quality of an individual slopes. Since this information is directly passed into the AAR, it is a good starting point (of course NODATA = 1 means that these points get a weight = 0 and are not passed into the AAR) for determining the statistical error for a single data entry.

The other statistical error, the one propagated currently in the PIPELINE is contained in the STDEV tag of the SPD. It is just the standard deviation of the slope of the fit of a 1st order polynomial to the integration ramp. In case of a perfect linear ramp this is a reasonable way of characterizing the quality of an individual slope. It provides additional information about the noise during the integration, which can be affected by instrumental effects or still some small glitches, not reflected in the GLITCH flag. From PIPELINE version 8.4 on this value is also passed straight to the AAR product without any conversion, therefore the units are in μV/second.

3 Systematic Errors

In order to treat systematic errors within the DAAR process and pass the STDEV straight to the AAR product a new IA structure (hidden in the PIPELINE processing) is used to transport the information. The AAR tag SPARE (BYTE Array [2]) is (mis-)used to store systematic errors in a byte code. The first array tag contains the "offset" error, the second array tag holds the "gain" error. Implementation details will be given in a specific chapter. As a baseline, both "offset" and "gain" errors are zero on entry of the DAAR stage.

4 DERIVE SPD

To calculate the slope in the SWS detector data, we take the AC-corrected detector outputs for every reset interval, y, versus time, t. Be aware that due to uncertainties in the AC-constants of every detector the slope might be slightly curved.
So the computed error is not suited to compare between detectors, moreover it represents the relative error of one detector.

$$y = a + bt \tag{1}$$

where a is the offset and b is the slope of one reset interval. As usual we write χ^2-equation as

$$\chi^2 = \sum (a + bt - y)^2 \tag{2}$$

We want to find a minimum in χ^2. To this end we differentiate the equation to a and b and put both equations to 0: the normal equations.

$$\frac{\partial \chi^2}{\partial a} = aN + b\sum t - \sum y \tag{3}$$

$$\frac{\partial \chi^2}{\partial b} = a\sum t + b\sum t^2 - \sum t \tag{4}$$

The solutions to this set of equations is

$$\Delta \equiv N\sum t^2 - \sum t \sum t \tag{5}$$

$$a = \frac{\sum t^2 \sum y - \sum t \sum ty}{\Delta} \tag{6}$$

$$b = \frac{N\sum ty - \sum t \sum y}{\Delta} \tag{7}$$

When we also form along with the other sums, the sum over y^2, $\sum y^2$, it can be shown that

$$\chi^2 = \sum y^2 - a\sum y - b\sum ty. \tag{8}$$

The variance of the solution equals to $\chi^2/(N-2)$ and hence the standard deviation is

$$\sigma = \sqrt{\chi^2/(N-2)} \tag{9}$$

N : number of samples

The formal errors in the offset and the slope are resp.

$$\sigma_a = \sigma\sqrt{\sum t^2/\Delta} \tag{10}$$
$$\sigma_b = \sigma\sqrt{N/\Delta} \tag{11}$$

The 'standard deviation of the slope', σ_b is passed into the SPD product and copied to the Auto Analysis product during the DAAR processing step.

5 DERIVE AAR

To understand the error handling of this part of the s/w it is unavoidable to mention some basics how the DERIVE AAR s/w is working. The DERIVE AAR part of the PIPELINE is noted step by step.

5.1 DARK : Dark Current subtraction

For non-darkcurrent blocks (e.g. source observations) a darkcurrent block before and after it with the same gain and reset length are taken. If both these darkcurrent blocks contains valid darkcurrents they are linearly interpolated in time and subtracted from the data block in between. In case only one of these two darkcurrent blocks contains valid darkcurrents it is subtracted from the block in between. If neither of the two darkcurrent blocks contains valid data the average darkcurrent for the entire SPD with the same gain and resetlength is subtracted. If there is no valid darkcurrent at all with the same gain and resetlength the block is flagged as having no data. Darkcurrents are never interpolated for a block that contains a photometric check.

For every dark current block a number which is used for error calculation is computed :

$$lo = index((total\_number + 1) * 1/4) \tag{12}$$
$$hi = index((total\_number + 1) * 3/4) \tag{13}$$
$$error = (dark(hi) - dark(lo))/2.0 \tag{14}$$
$$\tag{15}$$

total_number	: total number of valid data points during dark block
index	: ordered indices of dark current data block
dark	: ordered array of slopes during dark current measurement
error	: a width value which is used for error propagation (error of the MEDIAN) - median dark error

The DARK error for an individual scan can then be expressed as:

$$SPARE(0) = \sqrt{error1^2 + error2^2 + \left(\frac{dark1 - dark2}{2}\right)^2} \tag{16}$$

with: $dark1$ = dark current before scan; $dark2$ = dark current after scan; $error1$ = median dark error of dark1; $error2$ = median dark error of dark2; units: [μV/s]

In case of only one valid dark current Bit 22 is set in the flag tag of the SPD/AAR to indicate an uncertainty of the dark subtraction and the result would be instead:

$$SPARE(0) = error \tag{17}$$

with: $error$ = median dark error of the valid dark current

5.2 RESPCAL

For each AOT band in the SPD the wavelength dependent responsitivities are normalized with respect to the key wavelength, which is done per detector. The signals in the SPD that belong to a certain AOT band are then divided by the normalized responsivities for that AOT band.

Additionally for an AOT 1 the responsivities are smoothed by doing a flux conserving interpolation using wavelength dependent factors which determine the range of wavelengths over which the smoothing has to be done. These factors depend both on AOT speed and on the AOT band as well as on wavelength.

The key information during this step is contained in CAL25. These calibration files contain an "error curve" (units: [%]) representing the accuracy of the RSRF including known spectral features. Those curves are identical to the ones derived for the EOM on RSRF ([Vandenbussche 199x])

Two error quantities have to be handled within RESPCAL per detector and scan. The DARK offset error and the error curve contained in CAL25. Division of DARK offset error by RSRF gives normalized absolute offset error:

$$SPARE(0) = \frac{SPARE(0)}{CAL25.DATA.RESPONSxx * (1 - CAL25.DATA.SIGMAxx)} \tag{18}$$

and SPARE(1) gets a copy of the CAL25 error:

$$SPARE(1) = CAL25.DATA.SIGMAxx \tag{19}$$

with:
$CAL25.DATA.RESPONSxx$ = CALFILE 25 response value linear interpolated conserving the flux
$CAL25.DATA.SIGMAxx$ = resulting error from linear interpolation conserving the flux. It is assumed the error follows the same integration formulae (trapezoid rule) as the fluxes, except that the error is divided by the square root of a dilution factor beeing the ratio between FWHM and the local step size in wavelength. Integrating over more wavelengths yields a smaller error while interpolating over a small part wavelength interval a larger error.

xx = number of detector within the selected CAL25; SPARE(0) contains now already normalized fluxes and SPARE(1) is still in [%]

Part of the uncertainty in the key wavelength numbers is caused by the RSRF uncertainty, however this is very difficult to disentangle. In order to get back the right key wavelength accuracies, the RSRF uncertainty at the key wavelength is subtracted first.

The errors can be written now as:

$$SPARE(1) = SPARE(1) - SPARE(1)_{at\ keywavelength} \tag{20}$$

This error computation is only applied for periods where a valid wavelength is associated. Therefore errors during dark current and photometric checks are passed through without any modification. This is important for the next processing step.

5.3 FLUXCON

The conversion from signal to flux is done by normalizing the signal of the photometric check by that in the calibration file. This value is multiplied by the ratio of flux over signal of an external calibration source. Finally the obtained conversion factors are applied to the SPD.

The important information to be included in the PIPELINE are the quoted accuracies of the SWS flux calibration at the key wavelengths for the respective PIPELINE version (see EOM on flux calibration ([Morris 199x]) or ([Salama et al. 1997]).
The REL_FLUX_ERRx tag of CAL13 contains the key wavelength accuracies (further on called "CAL13error"). We know that for some AOT bands (especially for Si:Ga bands) we have even flux dependent errors (larger errors in the high flux regime). Therfore the calfile 13 structure contains error for low, median and high flux regimes. The boundaries of these regimes are determined by comparing any individual data point against limits given in the calfile 13 header (Keyword : ERRLOW and ERRMED).

Two more sources of error have to be handled within FLUXCON, the DARK error of the photometric check and the noise of the photometric check. The later one has been derived already in the current PIPELINE, the error due to the dark current in front of the photometric check has been neglected so far.

$$SPARE(1) = \sqrt{(SPARE(1) + CAL13error)^2 + photerrnoise^2 + photerrdark^2} \tag{21}$$

with:
$photerrnoise$ = noise of the photometric check data. Calfile 41 contains the representative shape of photometric checks. The actual photometric check is mapped onto the representative photometric check data using the relation of the MEANs as multiplication factor. The square root of the difference of actual photometric check data and representative photometric check data divided by the number of points is used to get the noise of the photometric check data.

$photerrdark$ = uncertainty originating from dark current of photometric check converted to relative units :

$$signal = MEDIAN3(photdata) \quad (22)$$
$$sigerr = MEDIAN3(photdata + darkdata) \quad (23)$$
$$photerrdark = MEDIAN(signal)/MEDIAN(sigerr) \quad (24)$$

with:
$MEDIAN3$ = 3 sample MEDIAN FILTER
$photdata$ = data of photometric check
$darkdata$ = dark current values of dark current preceeding the photometric check

The SPARE(0) value has to be converted first from normalized to absolute [Jy] in the same way the fluxes are by running through FLUXCON. Then:

$$SPARE(0) = SPARE(0) * (1 + CAL13error) \quad (25)$$

5.4 Implementation

Within IA error data are transfered between separate PIPELINE steps using the IDL structure error which has to be given as KEYWORD. This error structure contains the tags OFFSET and GAIN which are synonym to the usage within this document.
During the step EXTRACT_AAR the error structure information is copied into the SPARE tags of the AAR product.
All this is invisible during real PIPELINE processing.
SPARE(1) contains the GAIN error in % ranging from 0 to 255.
SPARE(0) contains the OFFSET value in Jy using a lookup table :

```
BYTE VALUE     Error       Function
-------------------------------------
  0    -     0.        byte * 0.1
  1    -     0.1
  .    .      .
 10    -     1.0
-------------------------------------
 11    -     2.0       byte - 9.
 12    -     3.0
  .    .      .
109    -   100.0
----------------------------
110    -   110.0       (byte-109)*10.+100.
111    -   120.0
112    -   130.0
```

```
  .      .       .
199    -  1000.0
------------------------------
200    -  1200.0       (byte-199)*200. + 1000.
201    -  1400.0
  .      .       .
255    - 11200.0   --> MAX value
------------------------------
```

For convenient decoding of errors it is possible to use *DECODE_AA_ERROR* of the SWS IA system or OSIA.

References

[Feuchtgruber 1998] Feuchtgruber, H., 1998, "Proposal for Modification of the Flux Errors in SWS Products"

[Kester 1996] Kester, D., 1996, "The calculation of slopes"

[Salama et al. 1997] Salama et al., 1997, ESA, SP-419, "Topics in SWS Data Analysis and Calibrations"

[Vandenbussche 199x] Vandenbussche, B., 199x, "End of Mission Report on SWS RSRF"

[Morris 199x] Morris, P., W., 199x, "End of Mission Report on SWS Flux Calibration"

6 Document change log

Version	Date	Initials	Comment
1.0	01 Mar 1997	EkW	First version
2.0	08 Jun 1999	EkW	Rewritten basing on the Proposal for Modification...
2.1	30 Aug 1999	EkW	Include remarks of K.Leech