Reducing AOT SWS07 Data in IA^{3*}

E. Sturm D. Kunze H. Feuchtgruber M. Thornley

November 14, 1997

NOTE: This document is subject to change due to ongoing IA/ISAP software development.

Before you begin:

This guide is intended to give a basic description of how to use IA tasks to reduce AOT7 data: it assumes that the FITS files containing the raw data have already been retrieved from the archive, and gives only a representative use of each command. This document provides background information for the accompanying IA script. For a full description of input parameters and alternate uses of a particular command, consult the IA help tool. This help tool is accessed by typing a "?" on the command line after starting an IA session. There are many possible help topics, so for each command described below, we have included, to the right hand side, the help topic where it can be found.

All IDL command syntax in an IA data reduction session is shown in **boldface** below, with variable and file names indicated in *italics*. All variable names are meant to be representative, but of course can be chosen by the user. These instructions suggest saving output after particularly time-consuming steps; the user should weigh the value of additional saves against the constraints of disk space. We also recommend that you work in a separate directory for each of your datasets, to prevent confusion.

Before starting IA, you should close all applications which use color (e.g., Netscape, Mosaic, or additional IDL sessions), as they will interfere with the IA color table. After IA has been started, you may restart these applications without affecting the IA session. In addition, for some tools a better performance is achieved if your window manager is set such that a window is active when the pointer is moved into it, but it is not automatically brought to the foreground. In general, it is advisable to keep the text window from which you started IA accessible, as some error messages from the interactive tools will be displayed in this window (note that some error messages will appear in pop-up windows of their own).

To begin: Start your IA session by typing **ia** at the system prompt. Then follow the instructions below.

1) **cal_select** (help: IA UTILITIES)

This tool is used to select calibration files. See Appendix B.4 of the IA User's Manual for descriptions of each calibration file. The cal_select command opens a window which lists all current calibration files, and there are two possible settings for each CAL file: 'BASE' or 'TEST'. The choice will depend on the current status of the calibration files, so ask an IA guru.

It is possible to load a cal_select.dat file (a binary file saved out of a previous run of cal_select) by clicking the 'load' button. NOTE: if using the 'load' option, the 'path' box must be filled in or cal_select will crash.

After either setting the appropriate CAL files manually or loading your calibration file, click 'OK' to exit the cal_select tool.

2) *erd*=**read_ferd**('xxxYYYzz') (help: IA INPUT OUTPUT)

This command reads in the ERD from the raw data files. The 8 digits in parentheses refer to the revolution number (xxx), target number (YYY), and observation sequence number (zz) from the archive listing. You may also use the command "read_fspd" to read in the SPD directly and skip to step 5; however, the pipeline reduction is ever-changing, and starting with the ERD allows you to take advantage of the most recent version of the reduction software.

3) *spd*=**dspd(***erd***)** (help: IA DERIVE SPD)

This command converts the ERD to an SPD, using the following CAL files: (1; 2; 2a,b; 3; 4; 5; 6; 12; 16a,b,c,e; 18; 23; 24).

4) *spd_orig*=**read_fspd('***SWSPxxxYYYzz.FITS***')** (help: IA INPUT OUTPUT)
spd.header=*spd_orig.header*
save,*spd***,file='***spd.save***',/xdr**

Unfortunately, the full header is not properly saved by dspd; the two commands in this step correct this problem by reading in the SPD (spd_orig) directly and copying its header into your recently derived SPD (spd). The filename given to read_fspd is in the format of files coming directly from the archive, where 'xxxYYYzz' are the same digits entered in step 2 above. The third command saves the SPD into an IDL save set, which can be restored in later sessions of IA if necessary. In that case, typing

restore,'*spd.save*' (help: ROUTINES)

will restore the IDL structure *spd* to your IA session.

NOTE: An SPD is rather large, and the use of the option /xdr makes the saving process significantly faster. The /xdr (for eXternal Data Representation) option is not necessary to save smaller quantities such as the sctbl or aotlr alone (e.g., step 6), but it is needed if you wish to export any save sets between VMS and UNIX machines (see the IDL help file on "save" for more information.)

5) **list_header,***spd* (help: IA UTILITIES)
set_plot,'ps
device,file='*aper_plot.ps*'**,/land**
aper_plot,*spd*,'5A',60. (help:IA UTILITIES)
aper_plot,*spd*,'5C',60.
aper_plot,*spd*,'6',60.
device,/close
set_plot,'x

These tasks give you more information about your dataset. The list_header command lists the FITS header, and in particular, you should note the position angle of your observations: the real position angle is (roll angle) + 90 degrees. The remaining commands in this step create a postscript file containing plots of the four SWS apertures on the sky (for bands 1 & 2; 3ACD; 3E; and 4). The value of 60. above is the declination scale of the plot in arcseconds. Consult your local system manager for printer options to determine how to print the postscript file locally.

6) *sctbl*=**make_sctbl(***spd***)** (help: IA UTILITIES)
print_sctbl,*sctbl***,file=**'*sctbl.txt*'**,/spawn**
save,*sctbl***,file=**'*sctbl.save*'

These commands create, print out, and save the scan table, which gives such information as the sequence of observations (darks, photometric checks, and science scans) and wavelengths covered by each band. The print_sctbl command prints the scan table to the screen as well as to the specified file name. The /spawn option is a local MPE command to print the specified text file to the default printer; again, consult your local system manager for printer options.

7) *aar*=**daar**(*spd*,**/all**) (help: IA AUTO ANALYSIS)
save,*aar*,**file**='*aar.save*',**/xdr**
status=**sap_wfits**('*aar.fits*',*aar*) (help: ISAP)

This task extracts an AAR from the reduced SPD, and the following two commands save the AAR as an IDL save set and a FITS file.

8) *index*=**where**(*aar.data.det* **le 48**)
aar_aot6=**cut_aar**(*aar, index*) (help: ISAP)
status=**sap_wfits**('*aar_aot6.fits*',*aar_aot6*) (help: ISAP)

If your F-P observation is merged with a grating observation (short wavelength section) it is recommended to split the AAR and to cut out the grating part for subsequent processing. Grating observations merged with F-P observations are always of type AOT6. The first command finds the index of all data belonging to detectors 1 to 48, i.e. to the grating part. F-P observations are made with detectors 49 (SW) and 51 (LW). Detectors 50 and 52 are redundant detectors which are not used for observations. The second command extracts a "sub-AAR", containing only the AOT6 part. The F-P part will be extracted in step 9. The last command saves the AOT6 AAR as a FITS file (e.g. for subsequent processing with ISAP).

9) *index*=**get_aotb_index**(*aar,'5B',stat*) (help: ISAP)
subind=**where**(*aar.data(index).wave* **ge 13 and $**
aar.data(index).wave **le 14**)
aar_fp=**cut_aar**(*aar, index(subind)*)

The order and aperture information in the scan table tells you which of the F-P bands 5A, 5B, 5C, 5D and 6 are present in your observation. See Table 1. If you have several different lines in one AOT band (see the scan table) you should further split the AAR. Subsequent processing is easier if there is a separate AAR for each spectral line. This splitting is done using the sequence get_aotb_index - where - cut_aar. In the above example first the index of all band 5B data and then the index of all data points of band 5B within the wavelength range (13-14 μm) is determined. Again "cut_aar" is used to extract a sub-AAR containing the desired data only.

Table 1: Relation between aperture, order, AOT band and wavelength for SWS07 detectors

Aperture	Order	AOT Band	Wavelength
1	3	5A	11.4 - 12.2
1	2	5B	12.2 - 16.0
2	2	5C	16.0 - 19.0
2	1	5D	19.0 - 26.0
3	1	6	26.0 - 44.5

10) **aar_fp.data.sdir=1**

The scan direction for F-P data is always 1, i.e. "up". Due to a software problem (which will be solved, soon) some of the valid F-P data sometimes are assigned a scan direction -1 ("down"), hence, would not be treated correctly in some of the following processing steps. To make sure that all data are included, simply overwrite the sdir tag of the AAR structure with a 1.

11) *flat_ref*=**sws_rebin(***aar_fp***,resol=10000,over=6,lines=0, $**
method='mean') (help:IA UTILITIES)

In this step a reference spectrum for the flatfielding in step 12 is created. There are several ways to do this. In the above example the reference is created through a rebinning of the data itself to a new grid of relatively low resolution. An alternative would be to use one of the scans which you consider a good one. In an AOT7 several scans of a line are counted as different lines, so you can select the scan via its associated line number:

good_line=**3**
index=**where(***aar_fp.data.line* **eq** *good_line***)**
flat_ref=**define_aar(data=***aar_fp.data(index)***)** (help:IA UTILITIES)

The "define_aar" command creates a new AAR, in this case containing only the line 3 data. (alternatively "cut_aar" could be used).

12) *aar_flat*=**sws_flatfield(***aar_fp***,2,ref=** *flat_ref***,/scaling)** (help:IA UTILITIES)
plot_aar,*aar_fp***,ps=1**
plot_aar,*aar_flat***,ps=2,/oplot**

Now the data are "flat fielded", i.e., to remove remaining differences in the relative flux calibration, the different scans are shifted to a common level. This common level is defined through the reference spectrum, which was created in the last step. The routine tries to fit polynomials of a given order (in this example it is 2) to the different scans. The correction is then done as shifting in an additive way (default) or as a scaling (used in this example). Scaling should be used only for strong sources where additive offsets due to uncertainties in the dark current subtraction can be neglected. The two plot commands allow you to check the result. First the original data are ploted (as + symbols) and then the flat fielded data are overplotted (as * symbols).

13) *aar_clip*=**sigclip(***aar_flat*,**resol=30000,lines=0,stop=0, $**
sigma=2.5,/nodata) (help:IA UTILITIES)
plot_aar,*aar_flat*,**ps=1**
plot_aar,*aar_clip*,**ps=2,/oplot**

Sigma clipping is recommended to get rid of outlyers and, to a certain extent, glitch tails. For each resolutin bin the median and standard deviation are calculated. Each point off by more than SIGMA standard deviations is flagged as NODATA (but not yet removed). By specifying the option "/nodata" all the corresponding data points are actually removed from the data set. The two plot commands again allow a check of the result like in the previous step.

14) *aar_reb*=**sws_rebin(***aar_clip*,**resol=30000,over=6,lines=0, $**
method='mean')
save,*aar_reb*,**file='***aar_reb.save***',/xdr**
status=sap_wfits('*aar_reb.fits***',***aar_reb***)**

A final rebinning can be used to average all scans into a single, smoother spectrum. The rebinning can be performed in various ways (mean, median, flux conserving, weighted or not, etc.). Please consult the on-line help for a detailed description of this routine. For the high F-P resolution there is not much difference between "mean" and "flux conserving".

Finishing up:

Type "help" to look at the list of IDL structures you have generated in this data reduction section. Make sure to save all files which were produced from the most time-consuming steps (i.e., any of the interactive tools) as IDL save sets and FITS files before exiting from IA. Then you can restore the SPD or AAR at any stage and redo some of the reduction if you find it necessary at a later date.