HAWC+ Polarimetry Pipeline Tutorial

This tutorial is designed to help guide users through the steps of processing HAWC+ polarization data, from raw data files (Level 0) to the final, calibrated (Level 4) data product. The information in this tutorial can also be found in more detail in the Data Reduction Pipeline (DRP) User's Manual.

Retrieve the Data

In this tutorial, we will be using polarimetry observations of NGC 2071 in Nod-Match-Chop (NMC) mode in band E.

To retrieve the data from the SOFIA Science Archive at IRSA, enter the following criteria in the main SOFIA search page (https://irsa.ipac.caltech.edu/applications/sofia):

Spatial Constraints → All-Sky

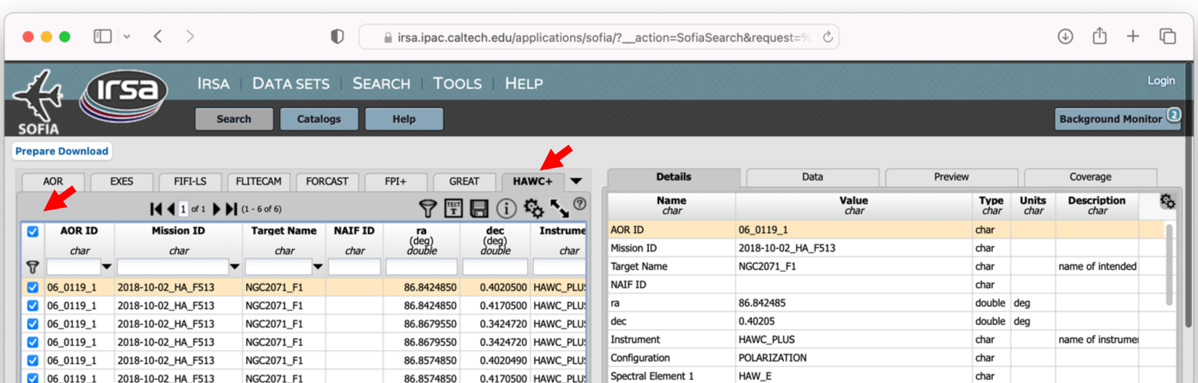
Observation Constraints → AOR ID → 06_0119_1

Data Product Constraints → Select Level 0 only (de-select Level 3 and Level 4)

or follow this link:

https://irsa.ipac.caltech.edu/applications/sofia/?api=search&spatialConstraints=allsky&execute=true&processingLevel=LEVEL_0&aorId=06_0119_1

On the search results page, click on the HAWC+ tab. On the HAWC+ tab, select all 6 data files, either individually or by selecting the box next to the AOR ID column header.



Click on Prepare Download to open a download dialog box, and then select Prepare Download within the dialog box. After a few moments, you will be asked to save a zip file to disk (about 2.43 GB in size). Save this file wherever you wish, unzip it, and navigate to the subfolders to locate the 6 raw HAWC+ FITS files. You may either leave these files in the directory or move them to another location of your choice; the folder containing the 6 FITS files will be referred to as the 'input' directory for the remainder of the tutorial.

The set of 6 files includes two calibration (CAL) files taken before and after a dither set of 4 polarization (POL) files, each containing 4 half-wave plate (HWP) positions.

Download and Install the Pipeline

Instructions on how to download and install the SOFIA data reduction pipeline package, *sofia_redux*, can be found at https://github.com/SOFIA-USRA/sofia_redux. If you need assistance, you can contact us through GitHub or the SOFIA Helpdesk (sofia_help@sofia.usra.edu).

Start the GUI

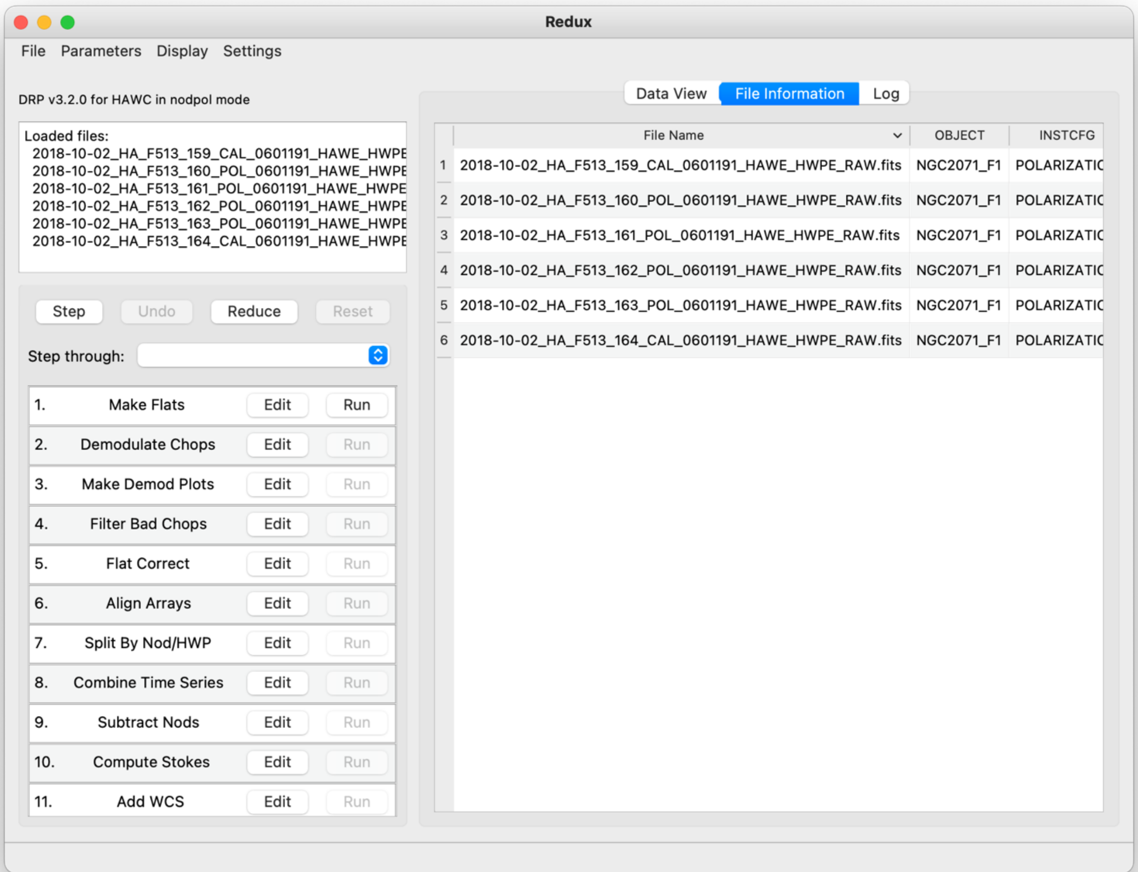
The pipeline can be executed either on the command line or through a GUI. In this tutorial, we will focus on running the pipeline through the GUI. To begin, open a terminal and navigate to a scratch directory.

Launch the GUI by typing:

```
> redux
```

The pipeline will write output files and logs to your current working directory, by default. If desired, the output directory may be modified from the File menu after a reduction is loaded.

In the Redux GUI, load the NGC 2071 data by selecting File → Open New Reduction, and navigating to the input directory where you saved the data. Select all 6 FITS files by clicking on the first file and shift-clicking on the last file, and then click Open. After clicking Open, the software reads the FITS headers and extracts relevant information, such as the instrument name and mode, to determine the appropriate pipeline to call to process the input data. Some basic FITS header information is shown on the "File Information" tab on the right side of the GUI.



A complete set of NMC polarimetric observations: four polarization observations at four half-wave-plate angles and two adjacent calibration observations.

When the pipeline runs, it is possible to automatically display the intermediate data using SAO DS9 if the connection between the pipeline/python and DS9 is set up properly (see http://ds9.si.edu/ for download and installation instructions; the ds9 executable must be available in the PATH environment variable for the pydsa9 interface to be able to find and control it). Alternatively, the automatic display can be disabled; each pipeline module can optionally produce FITS file output in case a different FITS viewer is preferred. Options for displaying images in DS9 are found under Display Settings on the Data View tab.

Run the Pipeline

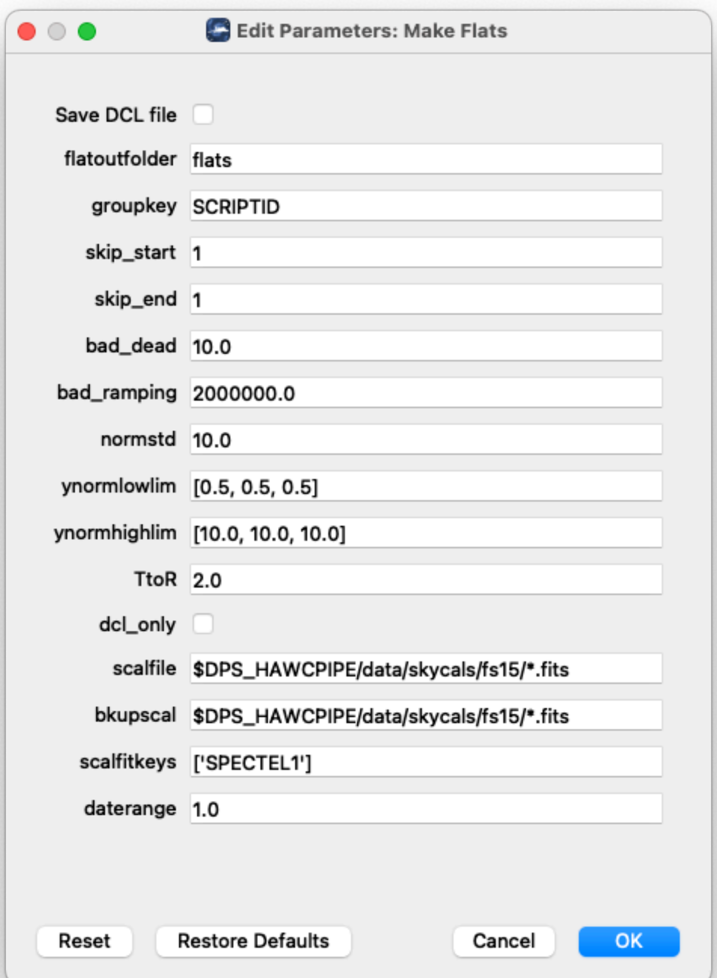
There are a total of 21 modules in the HAWC+ polarimetry pipeline in Nod-Pol mode, shown in the bottom-left panel of the GUI, but not all of them are user configurable or make significant changes to the data. We will focus on those modules that may require user interaction.

Each pipeline module can be configured by clicking the ‘Edit’ button for that module. The entire pipeline can be executed by clicking the ‘Reduce’ button. Alternatively, the modules can be run individually by clicking on the ‘Run’ button for that module, or in a small group by selecting an end point in the ‘Step Through’ dropdown menu, then clicking ‘Step.’ In this tutorial, we’ll run the modules one by one and examine their output.

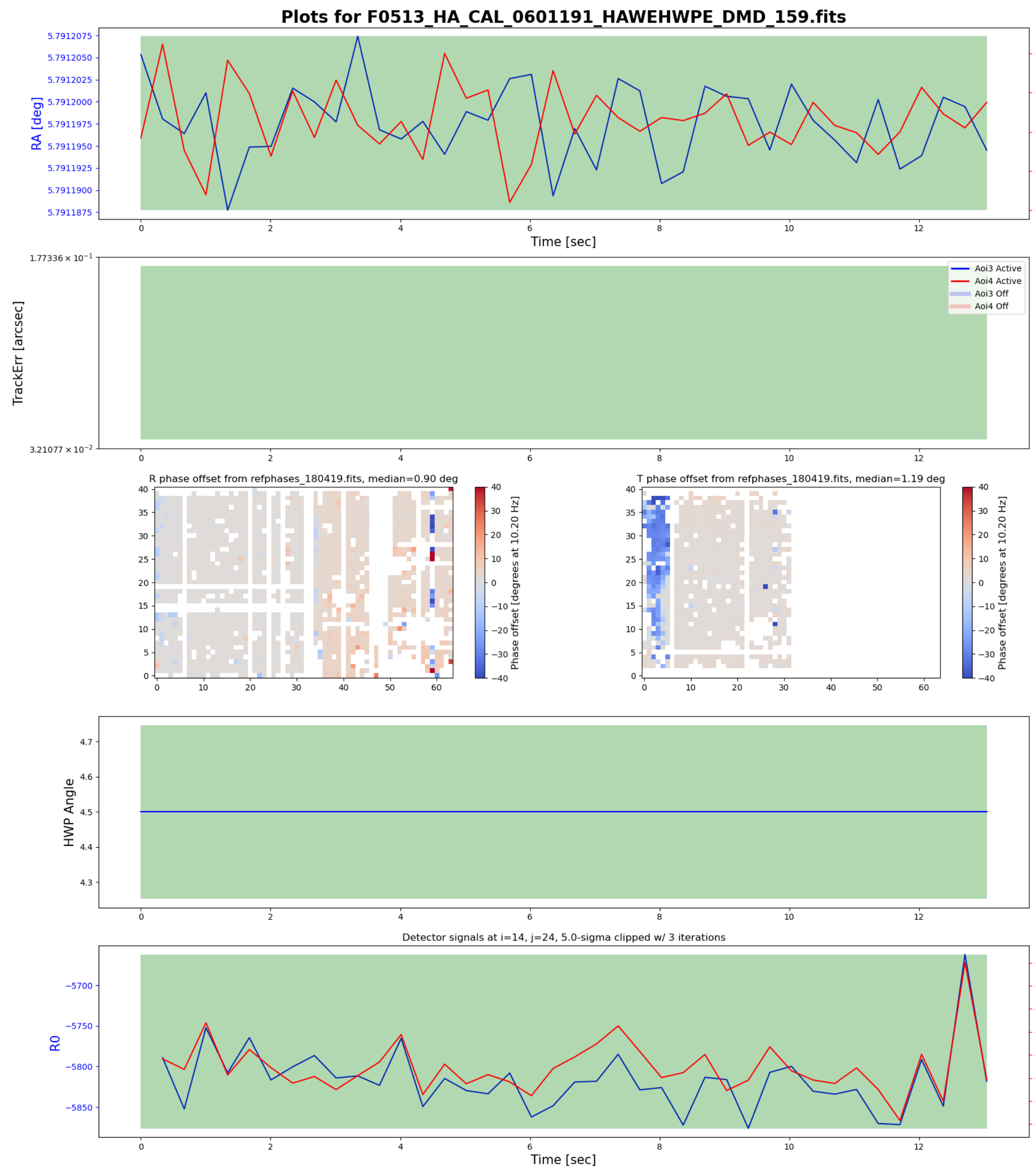
For a list of all the intermediate and final products and whether they are saved by default, see the tables in the data products section of the User’s Manual.

1. Make Flats

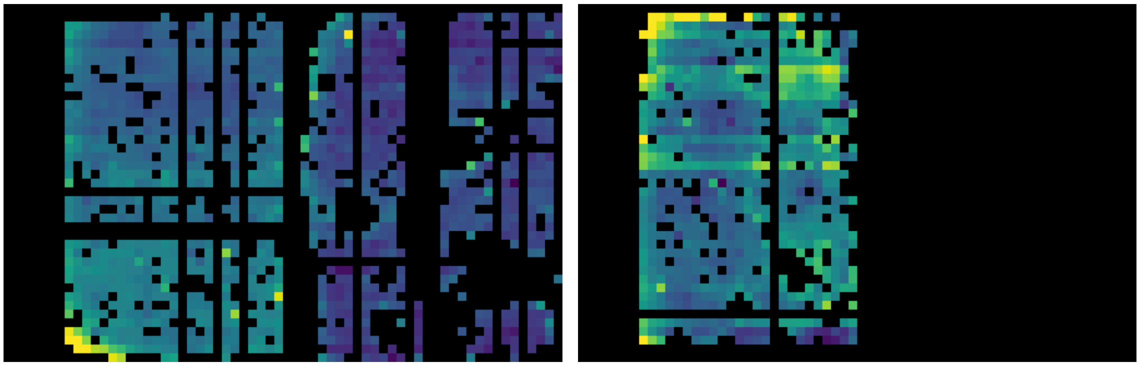
As part of preparing Level 0 data for reduction steps, the pipeline generates the flat fields in the Make Flats step from the input CAL files, using a Sky Cal file packaged with the pipeline for reference.



Output: The result of the flat-making process is an ‘obsflat’ (OFT) FITS file saved to a ‘flats’ folder on disk. The file contains R ARRAY GAIN and T ARRAY GAIN extensions, representing normalized flat field corrections for detector gain variations in the current observation. Additionally, a diagnostic plot is saved to disk for each CAL file, showing the average phase variations in the calibration files (*DPL*.png).



Diagnostic plot F0513_HA_CAL_0601191_HAWEHWPE_DPL_159.png, showing normal phase offset values (median < 5 degrees).



Obsflat F0513_HA_CAL_0601191_HAWEHWPE_OFT_159-164.fits with R array gains (left) and T array gains (right). Bad or missing pixels are marked NaN (black).

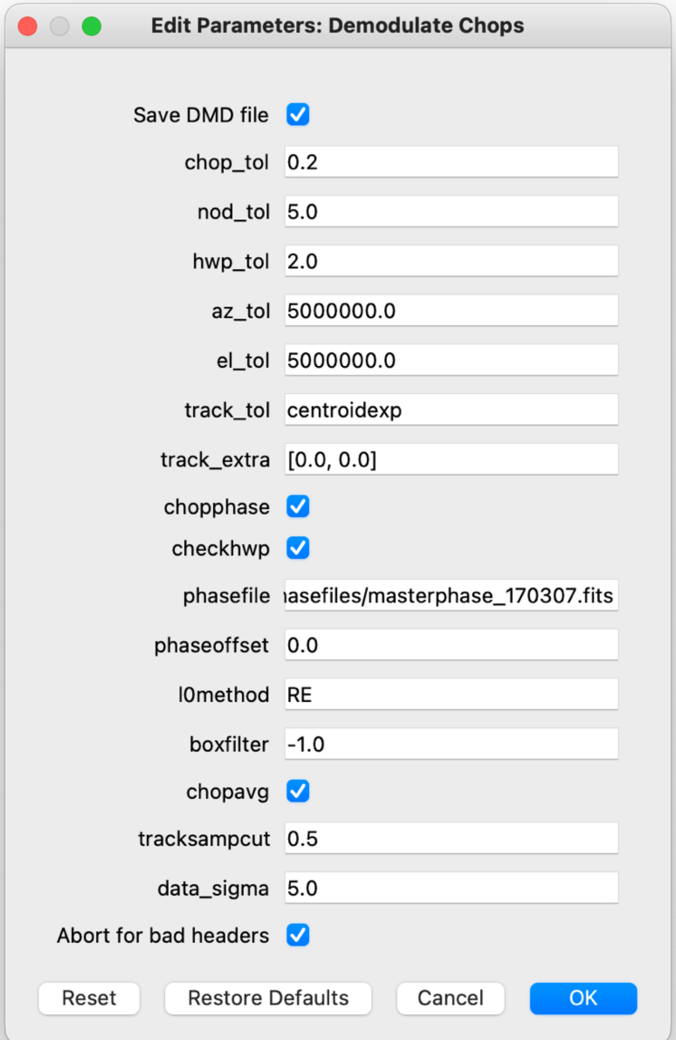
2. Demodulate Chops

The main purpose of this step is to subtract chop positions and identify the usable data by applying some filters.

By default, the raw data is first filtered with a box high-pass filter with a time constant of one over the chop frequency times the sample frequency. Then, any data taken during telescope movement (line-of-sight rewinds, for example, or tracking errors) is flagged for removal. Finally, the timestream data is weighted with a sine curve at the chop frequency and averaged at each nod position, effectively subtracting low-chop data from high-chop data and ignoring in-between states.

Most parameters for this step are for expert use only and will not need user modification. Occasionally, the diagnostic plot produced in the Make Flats step (above) shows a larger than average phase offset (> 5-6 degrees). To correct for it, you may need to enter the median phase offset value in the 'phaseoffset' parameter for the Demodulate Chops step.

In this case, the median phase offset is around 1 degree and does not need correction.

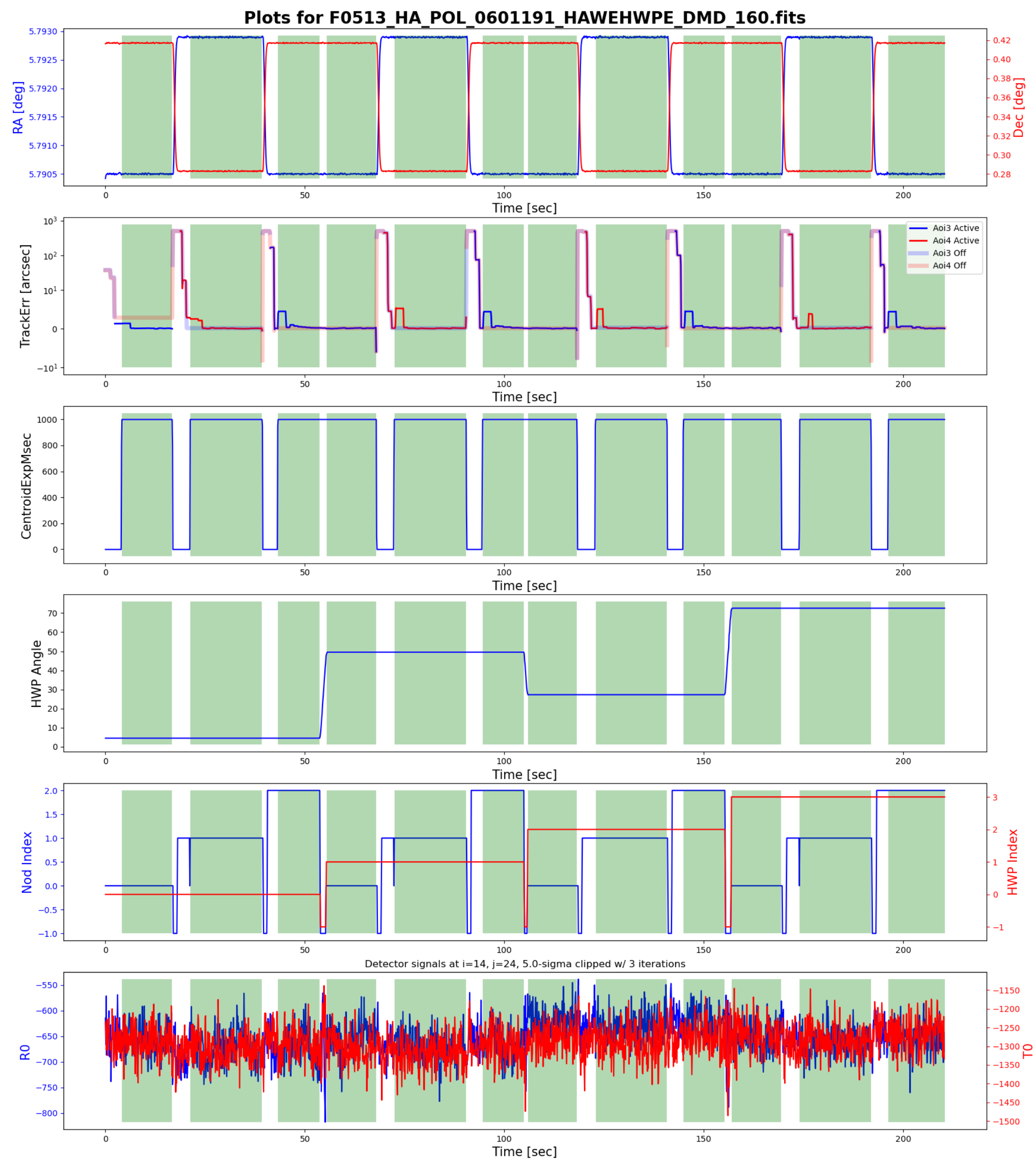


Output: The result of the demodulation process is a chop-subtracted flux value and associated variance for each nod position, HWP angle, and detector pixel. The output is stored in a new FITS table, in the extension called DEMODULATED DATA, which replaces the original TIMESTREAM data extension.

3. Make Demod Plots

This step makes some diagnostic plots for science data showing various traces of interest in the raw timestream. Anomalies in the diagnostic plots may indicate poor tracking or vignetting within the observation, potentially impacting the data quality of the observation. Poor tracking may appear in the plots as large regions not marked green (good) along with discrepancies in the CentroidExpMsec or TrackErr traces. Vignetting may appear as disruptions in the R0 and T0 flux trace at the bottom of the plot; it may also be directly flagged for the user with a warning in the plot itself. It is good to examine these plots for potential data quality issues, but this observation shows nominal plots for all input files, so no further action is needed.

hwp	nod	R_ratio	T_ratio	nod	R_ratio	T_ratio	nod	R_ratio	T_ratio
0	0	0.936	0.965	1	0.915	0.950	2	0.904	0.939
1	0	0.918	0.884	1	0.912	0.969	2	0.899	0.941
2	0	0.938	0.931	1	0.925	0.996	2	0.878	0.944
3	0	0.897	0.975	1	0.926	0.966	2	0.918	0.927



Diagnostic plot F0513_HA_POL_0601191_HAWEHWPE_DPL_160.png, showing normal tracking and data flagging.

4. Filter Bad Chops

This step removes previously flagged bad data from the timestream (marked white in the diagnostic plots). In most cases, users will want to use the default values in this module.

Output for this step has the same format as the previous step and is not typically saved to disk.

5. Flat Correct

In this step, the pipeline corrects the data for pixel-to-pixel gain variations by applying the flat field correction generated in the first pipeline step. Parameters for this step should not need modification unless an alternate flat is desired.

Output for this step is reformatted from a binary table to multiple FITS image extensions, containing R and T array flux values and associated variances and bad pixel masks. The array data is stored as a data cube, where each time sample is a separate frame at varying nod positions.

6. Align Arrays

In this step, R and T arrays may optionally be aligned to each other with small rotations and displacements. This correction is not typically used: the default parameters are set such that the step has no effect.

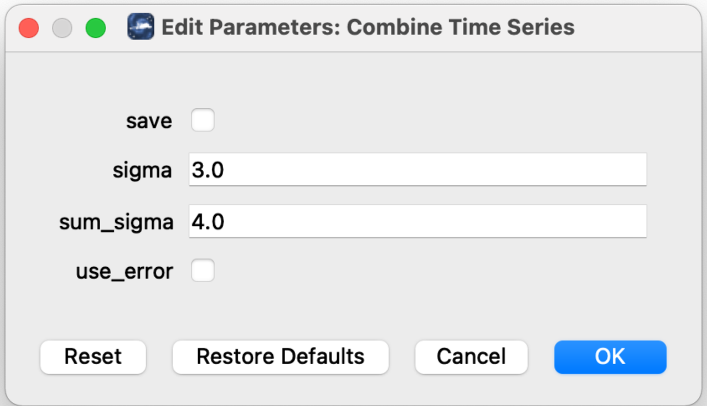
7. Split By Nod/HWP

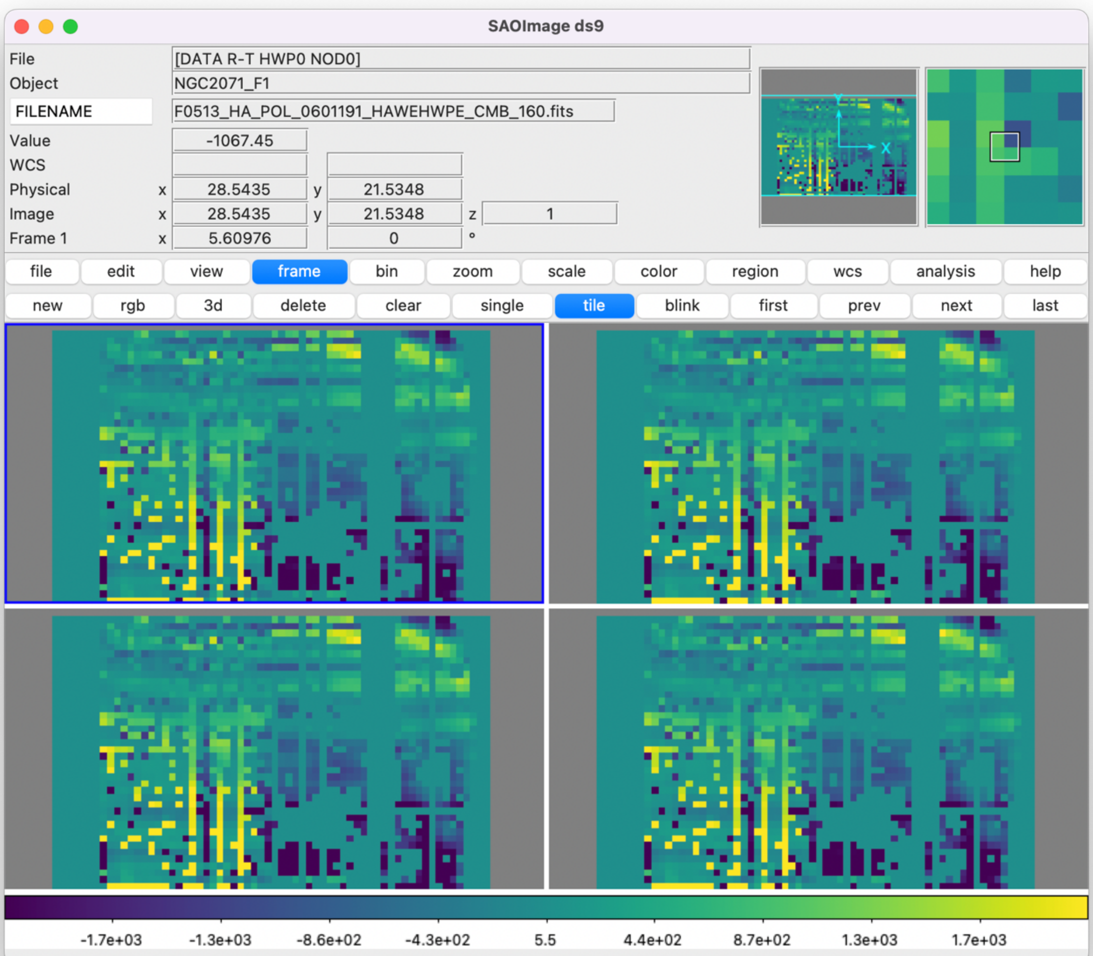
The pipeline next splits the data into separate image extensions for each nod position at each HWP angle, calculates the sum and difference of the R and T arrays, and merges the R and T array bad pixel masks. Parameters for this step should not need modification.

8. Combine Time Series

The pipeline combines all chop cycles at a given nod position and HWP angle by computing a robust mean of all the frames in the R+T and R-T images, rejecting pixels with more than 3-sigma difference from the mean value, by default.

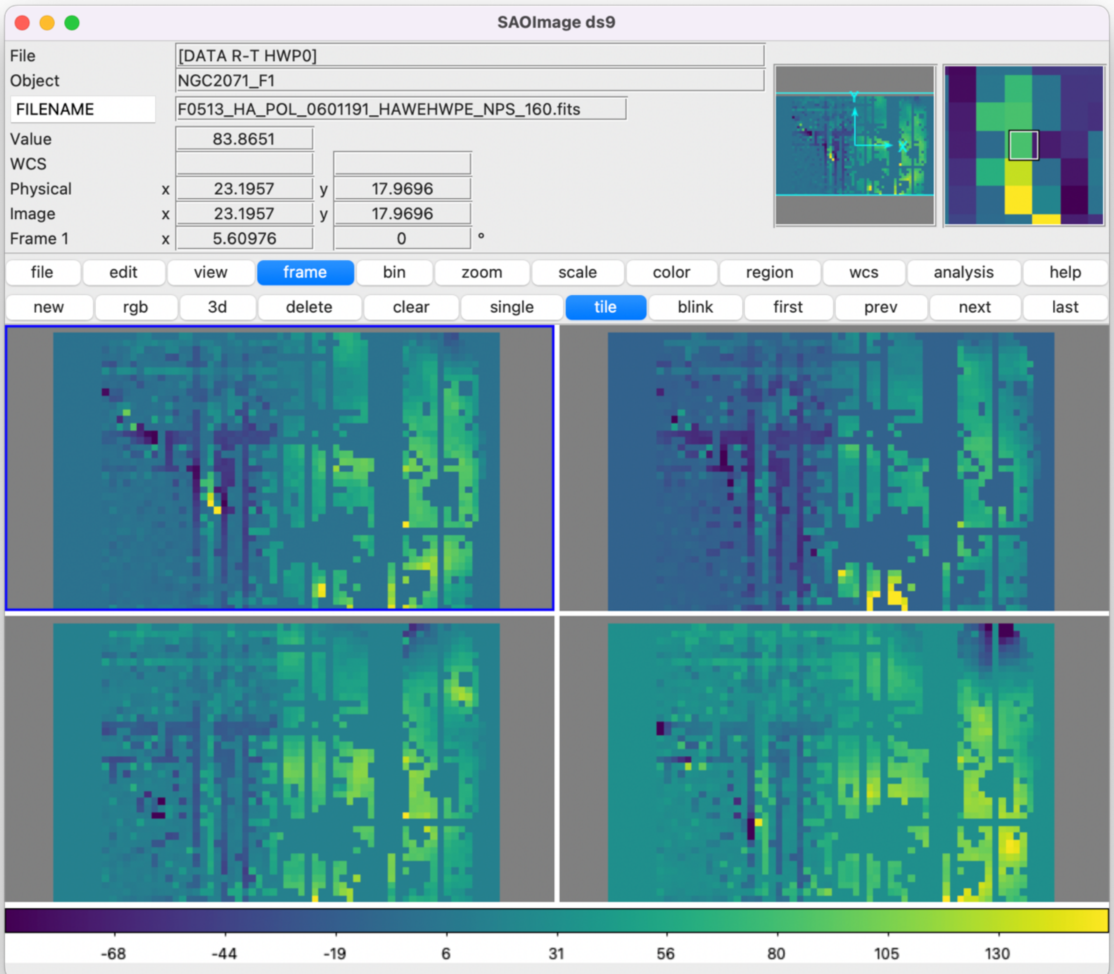
Output images from this step are now 2-dimensional, since the timestream has now been fully combined. The data still contains R-T and R+T extensions for each nod and HWP position.


Edit Parameters: Combine Time Series
save
sigma 3.0
sum_sigma 4.0
use_error
Reset
Restore Defaults
Cancel
OK


SAOImage ds9
File [DATA R-T HWP0 NOD0]
Object NGC2071_F1
FILENAME F0513_HA_POL_0601191_HAWEHWPE_CMB_160.fits
Value -1067.45
WCS
Physical x 28.5435 y 21.5348
Image x 28.5435 y 21.5348 z 1
Frame 1 x 5.60976 0 °
file edit view frame bin zoom scale color region wcs analysis help
new rgb 3d delete clear single tile blink first prev next last
-1.7e+03 -1.3e+03 -8.6e+02 -4.3e+02 5.5 4.4e+02 8.7e+02 1.3e+03 1.7e+03

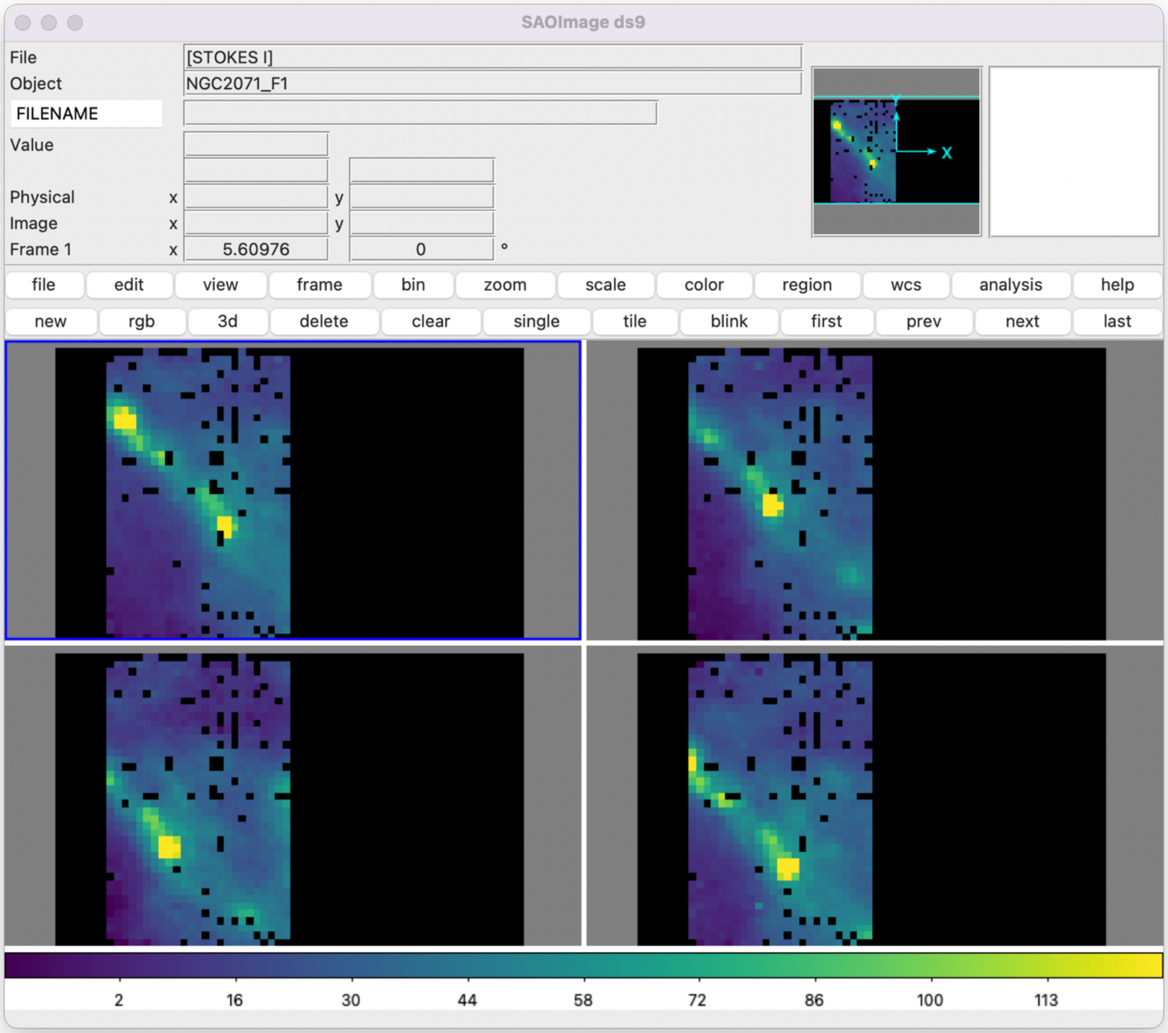
9. Subtract Nods

Here, the sky nod positions (B beams) are subtracted from the source nod positions (A beams) at each HWP angle and for each set of R+T and R-T, and the resulting flux is divided by two for normalization.



10. Compute Stokes

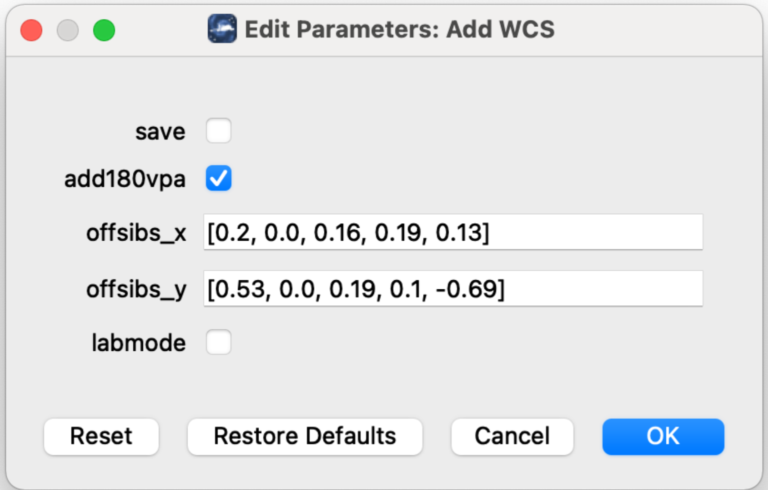
From the R+T and R-T data for each HWP angle, the pipeline now computes images corresponding to the Stokes I, Q, and U parameters for each pixel.

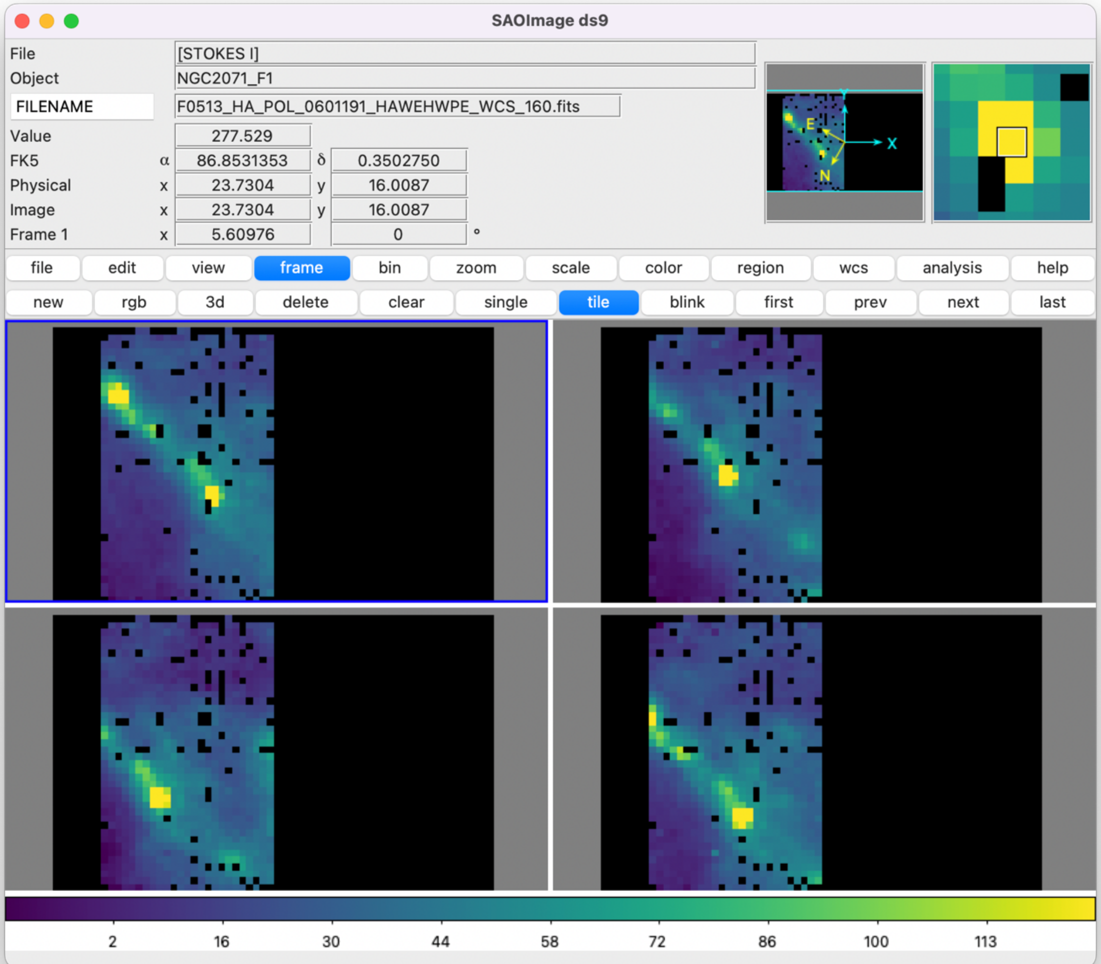


11. Add WCS

The pipeline uses FITS header keywords describing the telescope position to calculate the reference right ascension and declination (CRVAL1/2), the pixel scale (CDELT1/2), and the rotation angle (CROTA2). It may also correct for small shifts in the pixel corresponding to the instrument boresight, depending on the filter used, by modifying the reference pixel (CRPIX1/2). These standard FITS world coordinate system (WCS) keywords are written to the header of the primary HDU.

Each flight series typically has a different set of offsets by filter to correct for minor optical shifts. These are pre-computed and retrieved by date and used to populate the offsibs_x and offsibs_y parameters for these steps. In this step, a correction of 0.13 pixels in the x-direction and -0.69 pixels in the y-direction is applied to the CRPIX1 and CRPIX2 values, respectively, for this Band E data.


Edit Parameters: Add WCS
save
add180vpa
offsibs_x
[0.2, 0.0, 0.16, 0.19, 0.13]
offsibs_y
[0.53, 0.0, 0.19, 0.1, -0.69]
labmode
Reset
Restore Defaults
Cancel
OK

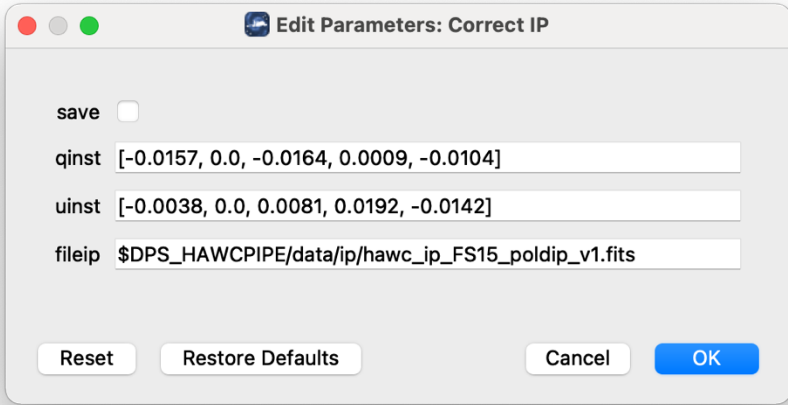


12. Correct IP

In this step, the pipeline attempts to remove the instrument and the telescope polarization that may introduce some foreground polarization to the data.

The instrument team uses measurements of the sky to characterize the introduced polarization in reduced Stokes parameters ($q = Q/I$ and $u = U/I$) for each filter band at each pixel.

For Nod-Pol data, a FITS file specifying q and u values by pixel is usually used to correct the instrumental polarization (the 'fileip' parameter). If a constant value should be used instead, 'fileip' should be set to 'uniform', and the values for each band in the 'qinst' and 'uinst' parameters are directly used.



13. Rotate Stokes

The calculated Stokes parameters reflect polarization angles measured in detector coordinates. After the foreground polarization is removed, the parameters may then be rotated into sky coordinates, using the recorded telescope vertical position angle (VPA) from the FITS header.

14. Correct Opacity

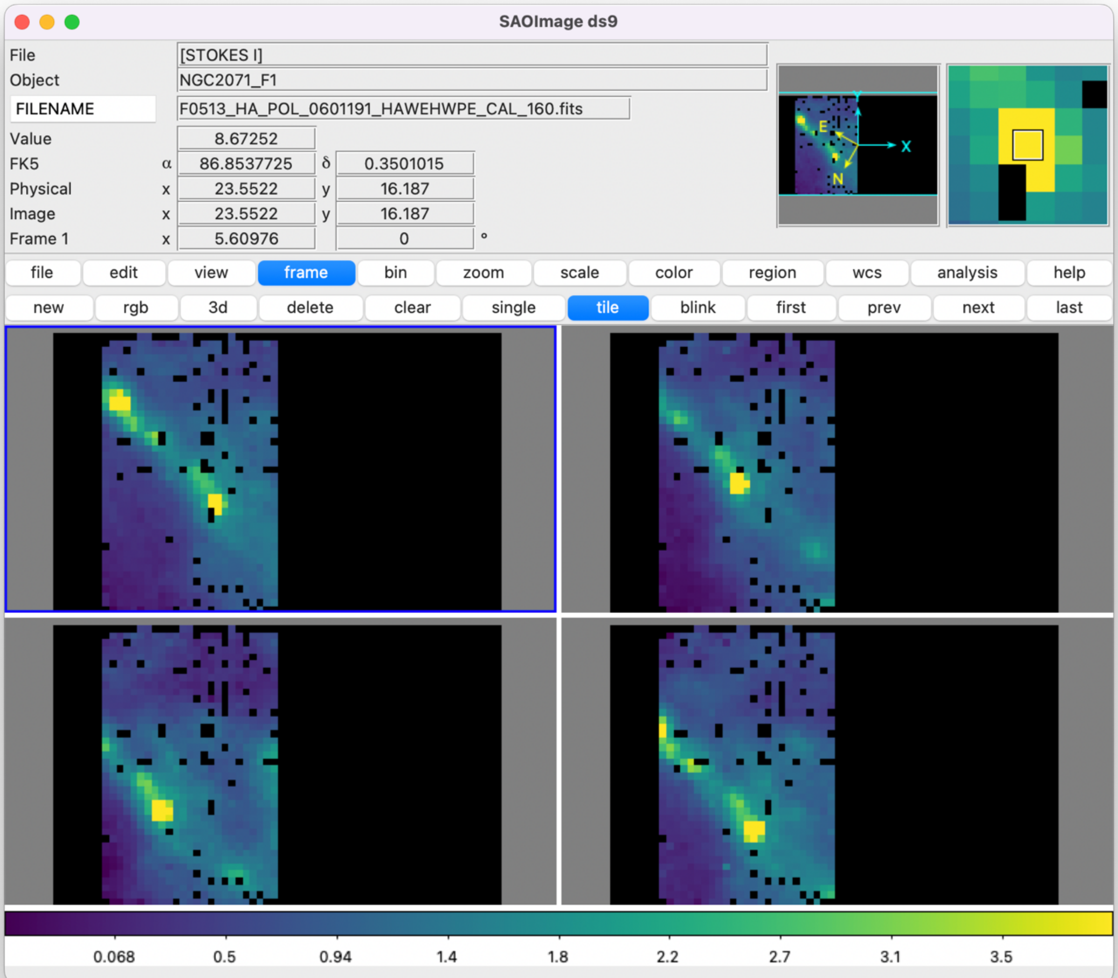
To combine images taken under differing atmospheric conditions, the pipeline corrects the flux in each individual file for the estimated atmospheric transmission during the observation, based on the altitude and zenith angle at the time when the observation was obtained.

Output: As the final step before flux calibration, this file is typically saved to disk with file code OPC. If recalibration is necessary, the early pipeline steps need not be repeated: OPC files can be directly loaded into Redux to complete a reduction.

15. Calibrate Flux

The pipeline now converts the flux units from instrumental counts to physical units of Jansky per pixel (Jy/pixel). For each filter band, the instrument team determines a calibration factor in counts/Jy/pixel appropriate to data that has been opacity-corrected to the reference zenith angle and altitude.

Output: The pipeline stores the flux calibrated map (CAL file) by default. The FITS extensions are unchanged: they contain STOKES I, STOKES Q, and STOKES U images with associated errors.



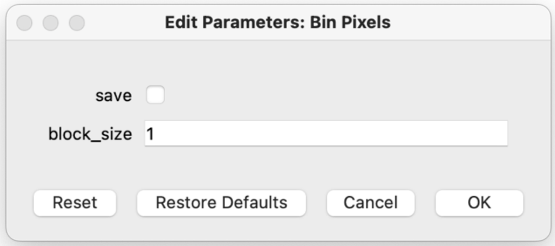
16. Subtract Background

In this step, the pipeline does a draft merge of all input maps to determine the average background level, then subtracts residual backgrounds from each input image to match the mean value. Parameters from this step do not typically need modification.

17. Bin Pixels

This step may be used to bin detector pixels together prior to merging maps, to decrease resolution and increase signal-to-noise in the final polarization vectors. To do so, set the block size to a value that divides the 40x64 array evenly (2, 4, or 8).

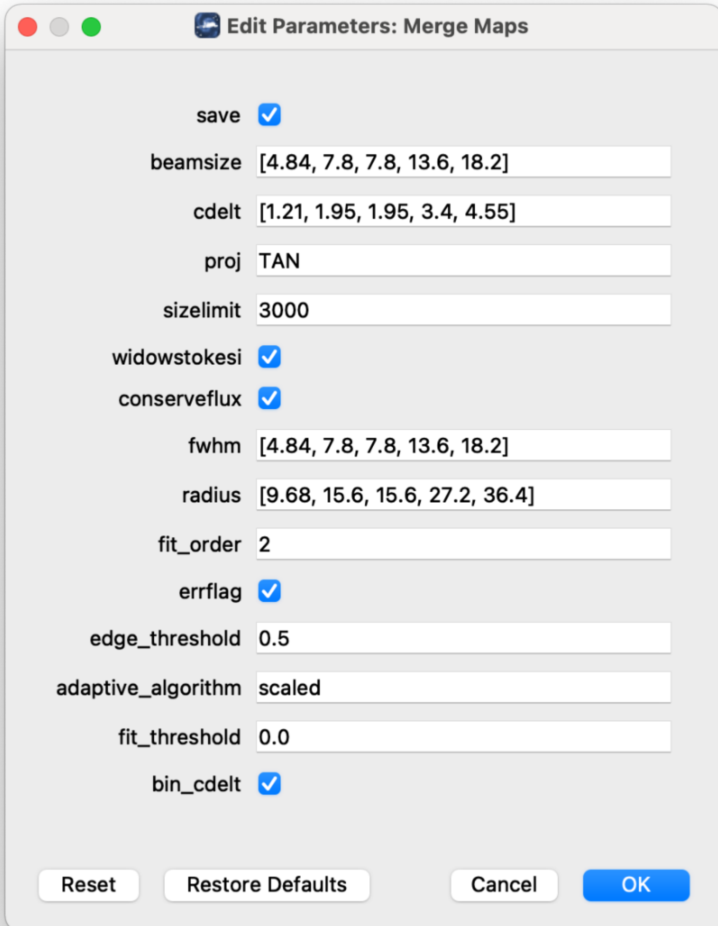
For this observation, we will use the default value of 1, which does not bin the detector pixels.

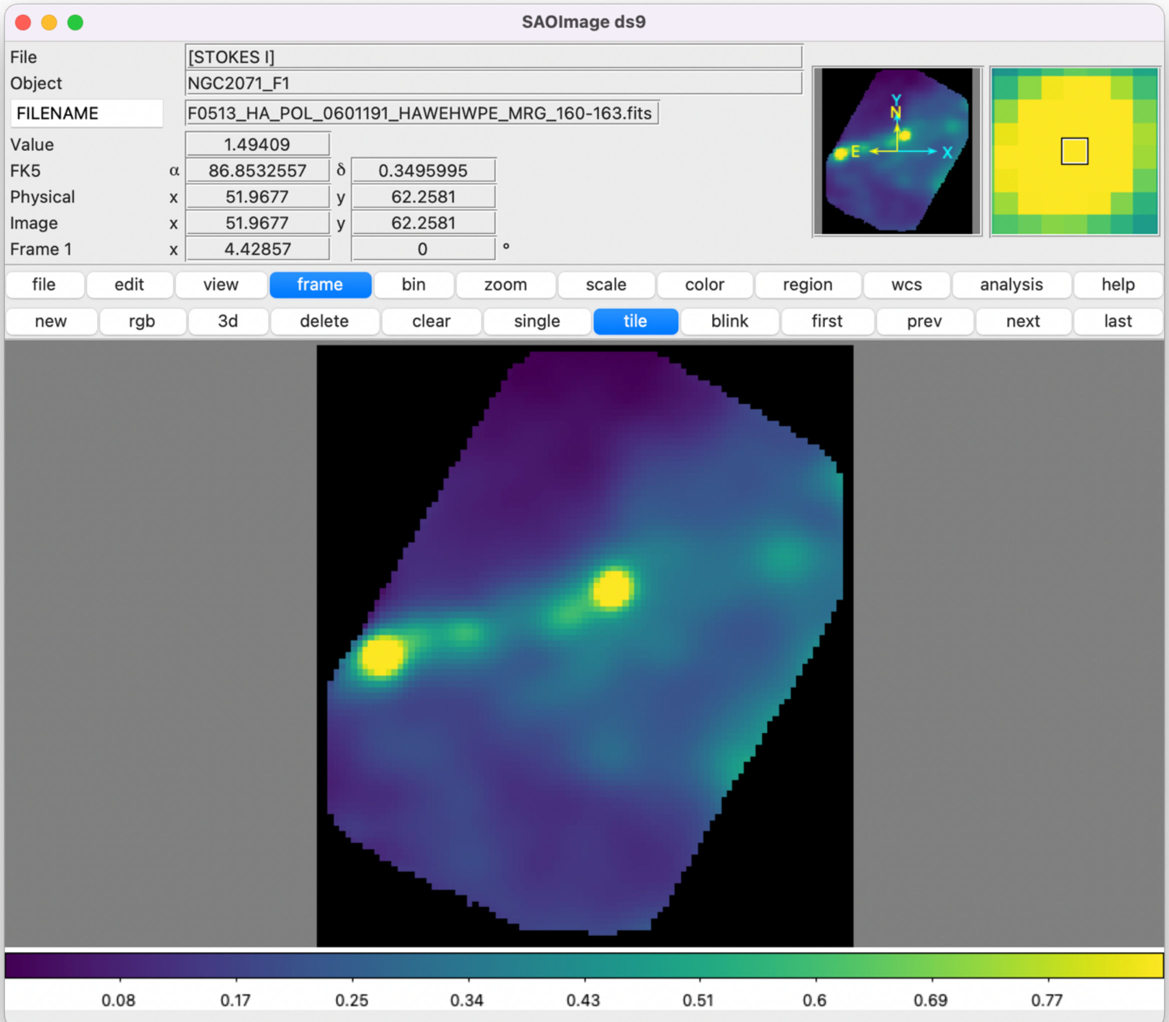


18. Merge Maps

Up until this step, each step was applied on a separate dither position. To merge the separate maps, the pipeline uses the WCS in each map to determine the sky location of all the input pixels. Then, for each pixel in the output grid, the algorithm considers all input pixels within a given radius that are not marked as bad pixels and averages them with a distance-weighted kernel.

The output grid size is set by the 'cdelt' parameter, with one value in arcsec for each filter band. For this observation in Band E, the output pixel size is 4.55 arcsec, corresponding to the beam size / 4. If larger output pixels are desired, to decrease resolution in the final map, set the final value in the cdelt parameter to a larger number.

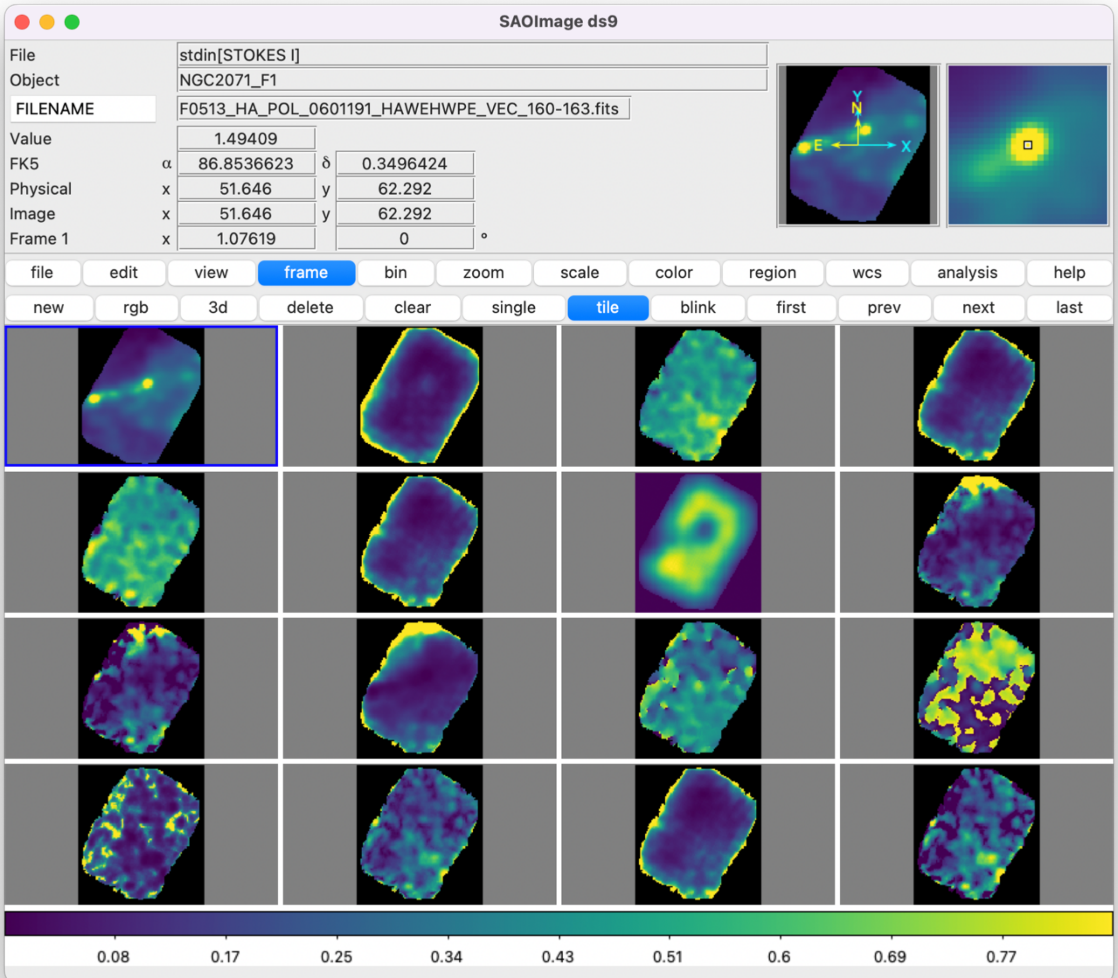

Edit Parameters: Merge Maps
save
beamsize [4.84, 7.8, 7.8, 13.6, 18.2]
cdelt [1.21, 1.95, 1.95, 3.4, 4.55]
proj TAN
sizelimit 3000
widowstokesi
conserveflux
fwhm [4.84, 7.8, 7.8, 13.6, 18.2]
radius [9.68, 15.6, 15.6, 27.2, 36.4]
fit_order 2
errflag
edge_threshold 0.5
adaptive_algorithm scaled
fit_threshold 0.0
bin_cdelt
Reset
Restore Defaults
Cancel
OK



Output: The output from this step is a single FITS file, containing a flux and error image for each of Stokes I, Q, and U, as well as the Stokes covariance images. An image mask is also produced, which represents how many input pixels went into each output pixel. Because of the weighting scheme, the values in this mask are not integers. A data table containing demodulated data merged from all input tables is also attached to the file with extension name MERGED DATA.

19. Compute Vectors

The polarization percentage (p) and angle (θ) and their associated errors (σ) are calculated using the Stokes I, Q, U images.



Output: The 19 extension FITS file has all the Stokes vector measurements and their uncertainties. 16 of the extensions (shown above) are images, containing Stokes parameter maps, percent polarization, polarization angle, polarized flux maps, and associated errors. Polarization vectors are additionally recorded in a POL DATA table extension.

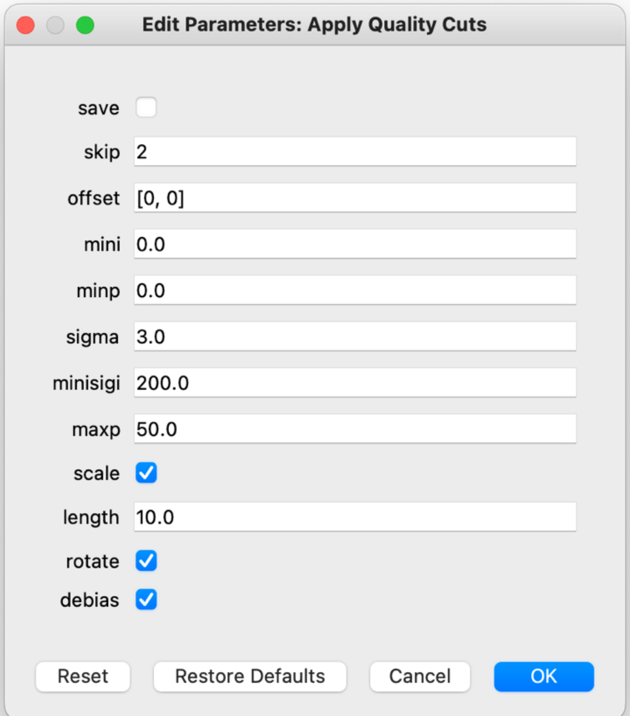
20. Apply Quality Cuts

In this step, data quality cuts are applied to the calculated vectors.

Several of these options may be useful for tuning the final output set of vectors:

- “skip”: allows you to keep vectors every ith pixel. The default of 2 represents Nyquist sampling as appropriate for cdelt=beamsize/4 in the merge step).
- “sigma”: rejects polarization values with error values greater than this many times the standard deviation (sigma).

- “minisigi”: (200 by default) rejects polarization vectors from pixels with Stokes I flux over the error on Stokes I (I/σ_I) less than this value.

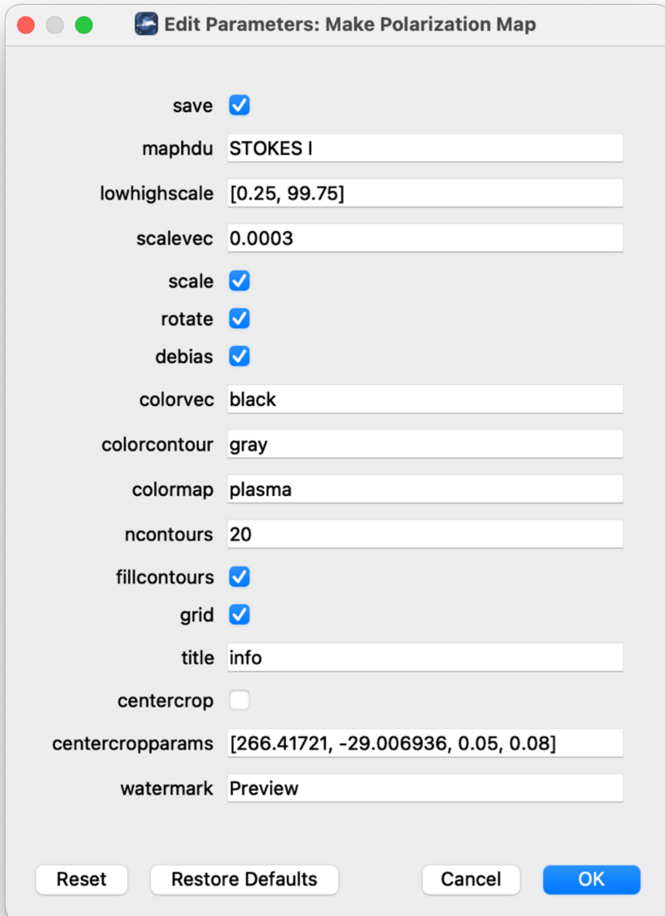


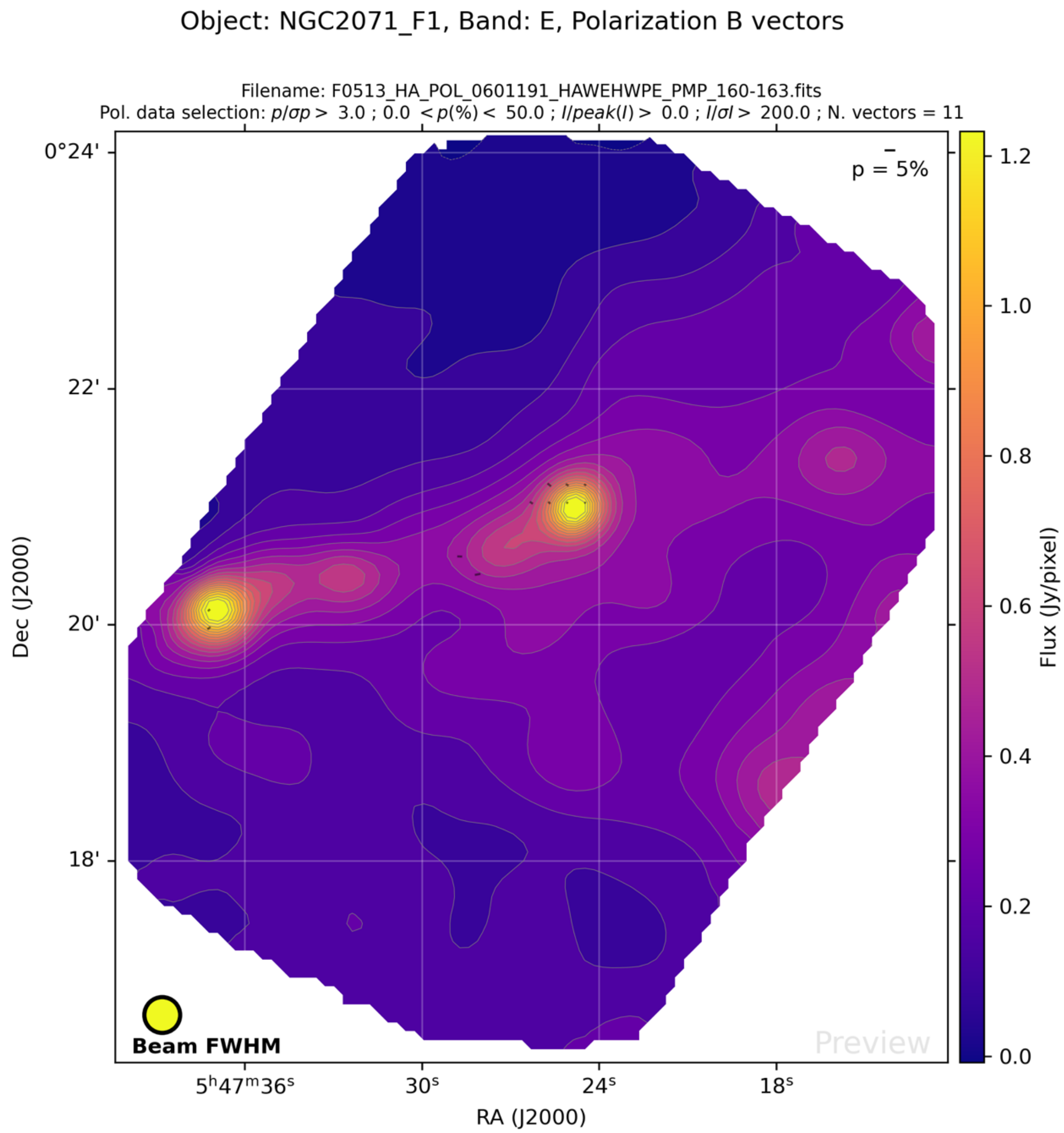
Output: Polarization vectors corresponding to all image pixels continue to be stored in the binary table extension called POL DATA. Polarization vectors that survive data quality cuts are stored in a new table extension called FINAL POL DATA. This table is used to generate vector region files and map images in the next pipeline step.

21. Make Polarization Map

The final step of the pipeline makes a preview image containing polarization vectors that passed through the applied quality cuts.

Output: The pipeline creates and stores the polarization map in a form of 18-extension FITS file. A PNG version of the map as well as a region file (.reg) that includes computed and rotated vectors that passed the quality cuts. This region file can be overlaid in DS9 (click on the “region” tab in ds9, then “open”).


Edit Parameters: Make Polarization Map
save
maphdu
STOKES I
lowhighscale
[0.25, 99.75]
scalevec
0.0003
scale
rotate
debias
colorvec
black
colorcontour
gray
colormap
plasma
ncontours
20
fillcontours
grid
title
info
centercrop
centercropparams
[266.41721, -29.006936, 0.05, 0.08]
watermark
Preview
Reset
Restore Defaults
Cancel
OK



Final polarization map, with 11 vectors surviving the data quality cuts.

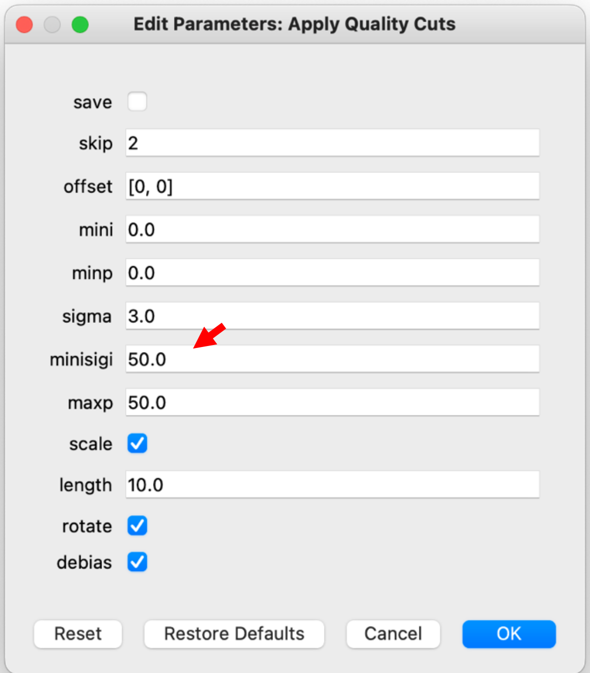
Revise quality cuts

By default, the pipeline allows only the most confident polarization vectors through to the final map. Relaxing the minisigi parameter (I/σ_I) to confident but lower values of I/σ_I, allows more polarization vectors through to the final map, which may be useful for some science cases.

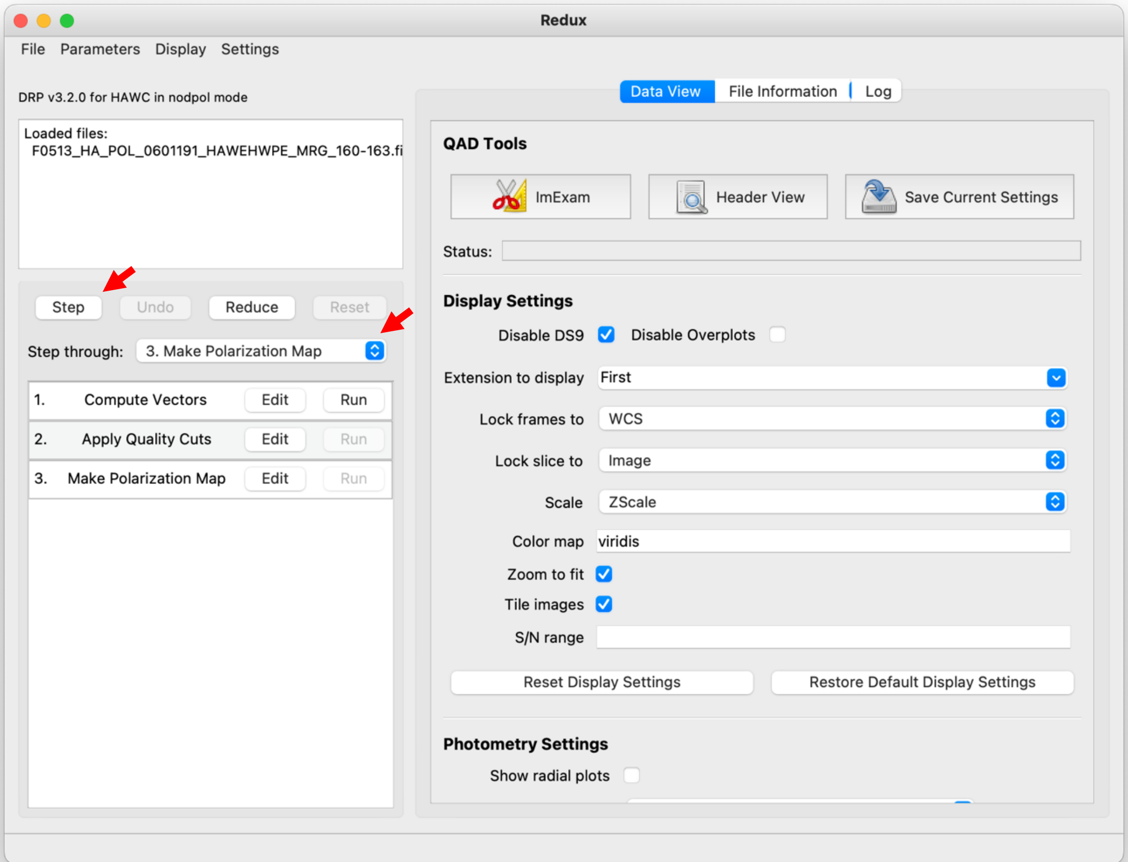
To make a new polarization map, load in an intermediate file produced at the Merge Maps step. Navigate to File → Open New Reduction, then select F0513_HA_POL_0601191_HAWEHWPE_MRG_160-163.fits from your output directory.

Redux, recognizing the intermediate data product, will load only the last few steps of the pipeline to run.

Click Edit next to the Apply Quality Cuts module and set minisigi to 50.0.

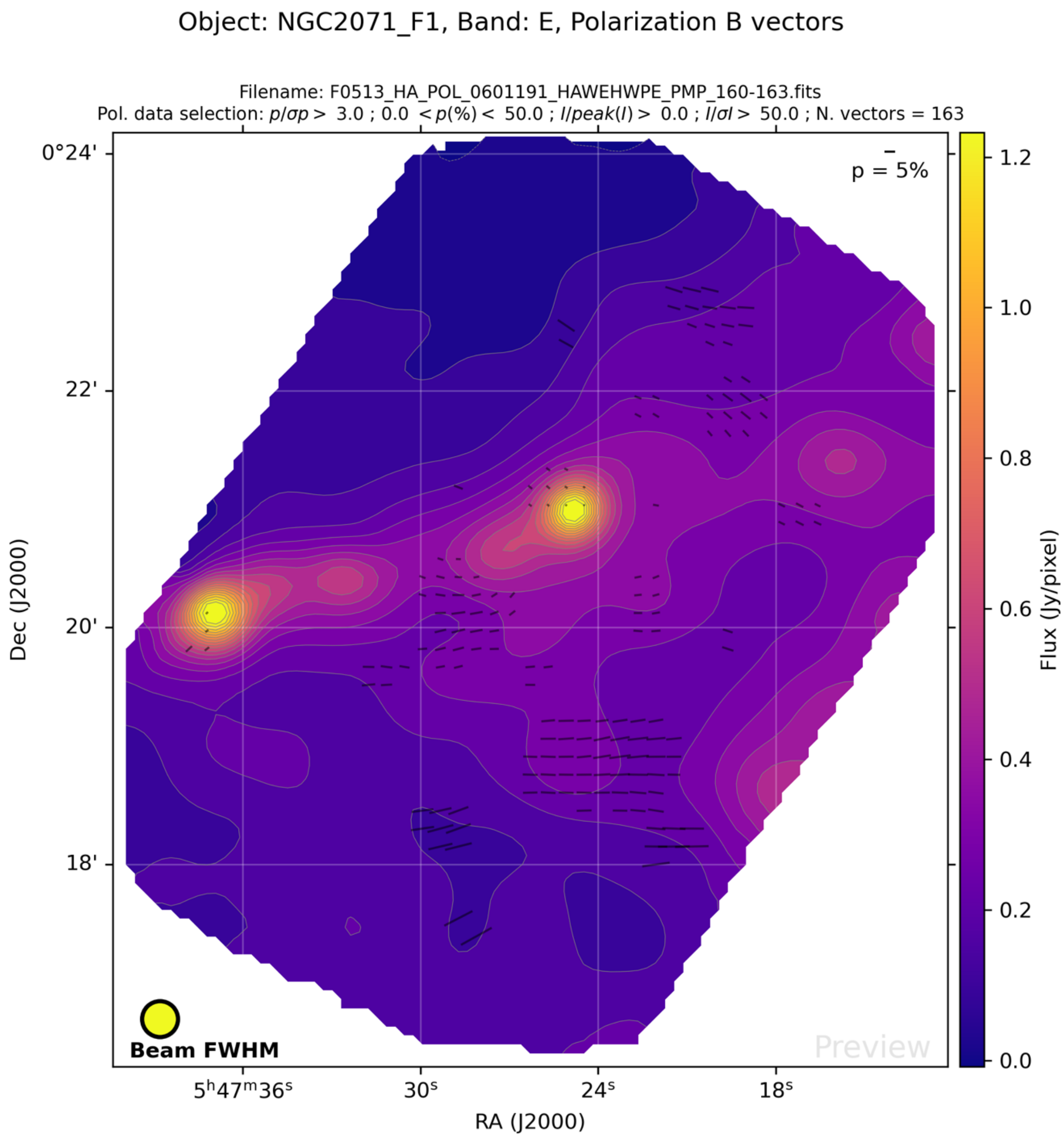


Select “Make Polarization Map” in the Step Through dropdown box, then click Step to run all three of the final pipeline steps at once.



A new polarization map is generated in the output directory, with many more vectors surviving the quality cuts.

To explore the effects of other quality cut parameters, click Undo, modify the parameters, and click Step again to repeat the reduction.



The final polarization map, with 163 vectors surviving quality cuts. Note that the data quality cut parameters are specified at the top of the image.

Quality Assessment

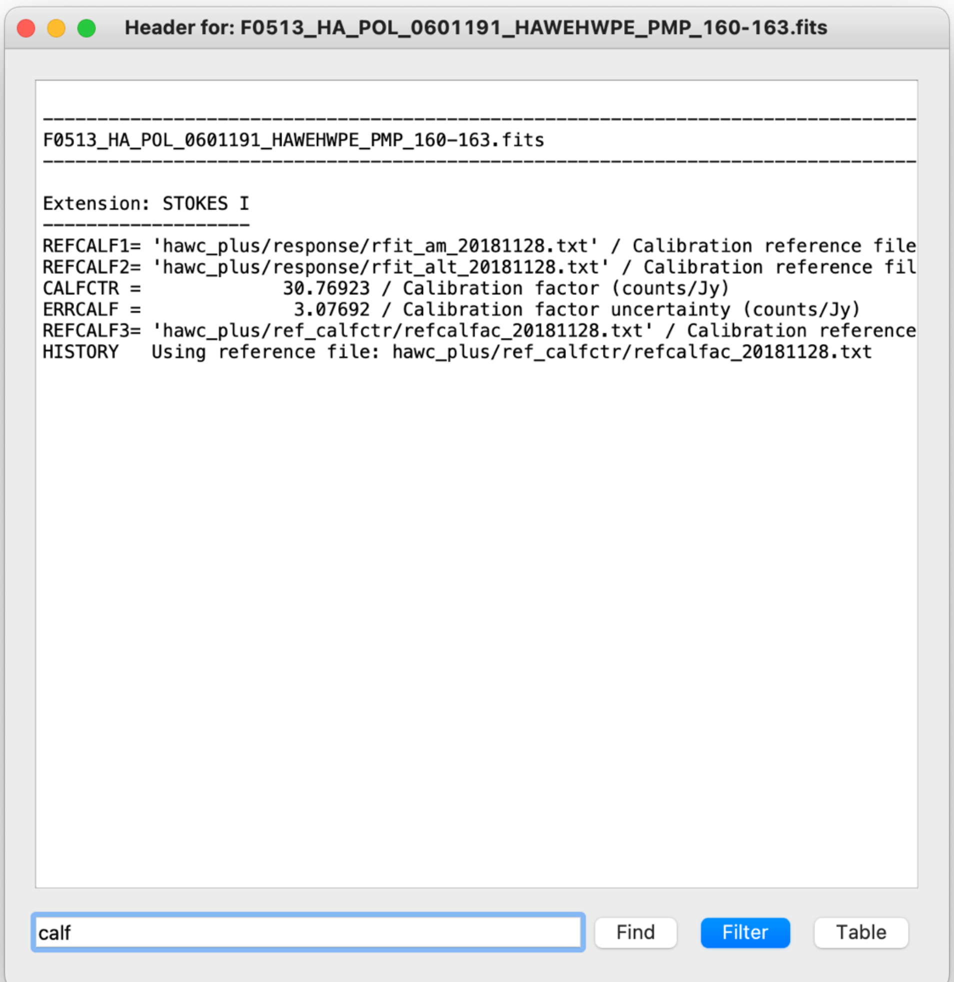
Once the final step of the pipeline is completed, the flux calibrated map is ready for quality checks. The Redux GUI displays images for **quality analysis and display (QAD)** in the DS9 FITS viewer.

- Check the data reduction log for ERRORs and WARNINGs. The log also lists the parameters and options that were used to run each module.

- Check calibration values for your data.

  Open the header by clicking on the Header View (magnifying glass icon) in the Data View tab of the Redux interface. Enter ‘calf’ in the filter text to see the applied calibration factor and associated references.

  The overall calibration accuracy is expected to be within about 10%. If the calibrated flux values are far from the expected values or discrepant with past observations of the same target with similar instrument configurations, users can track the specific calibration files that were used for their data and examine or update the calibration factor independently.



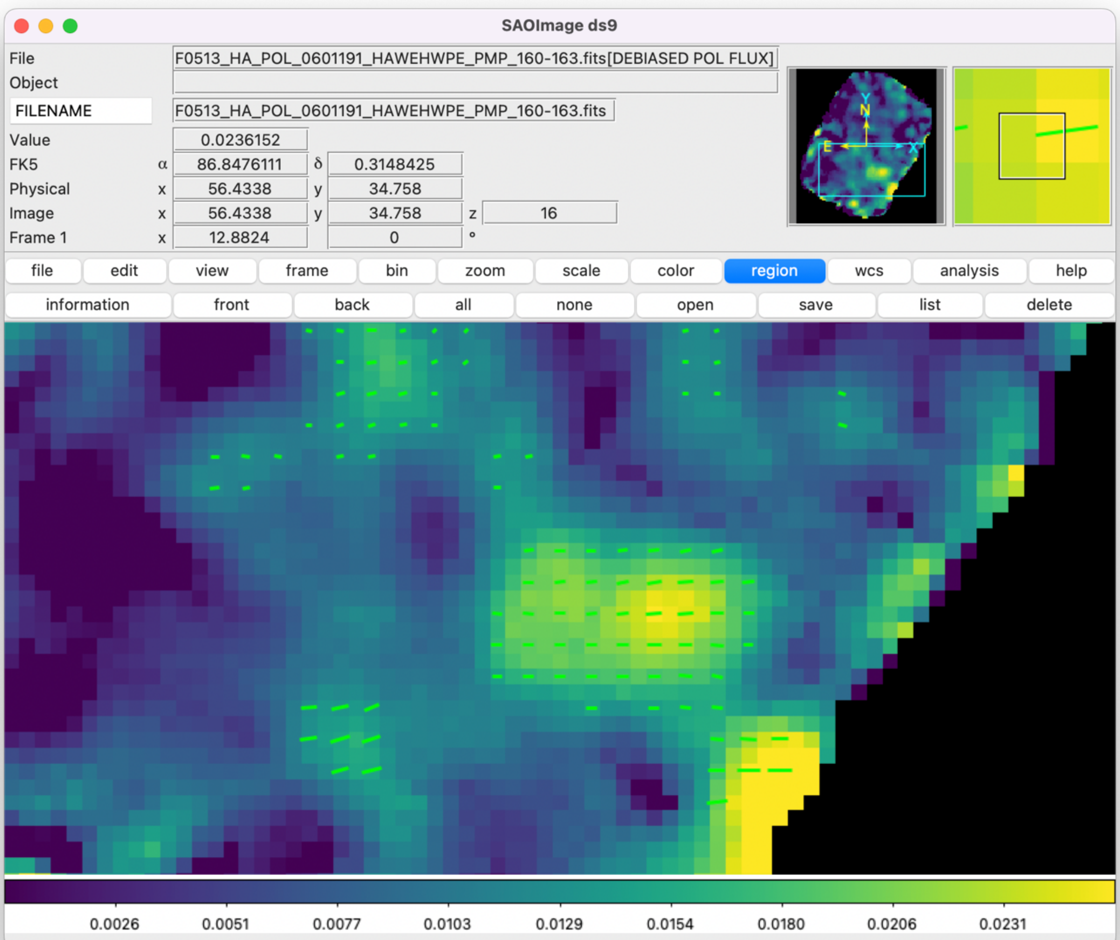
- Check the PSF shape for the source.

The QAD tools include some photometry and image inspection tools, via the ImExam button (scissors icon). These are described more fully in the HAWC+ Imaging tutorial or in the User's Manual for the pipeline.

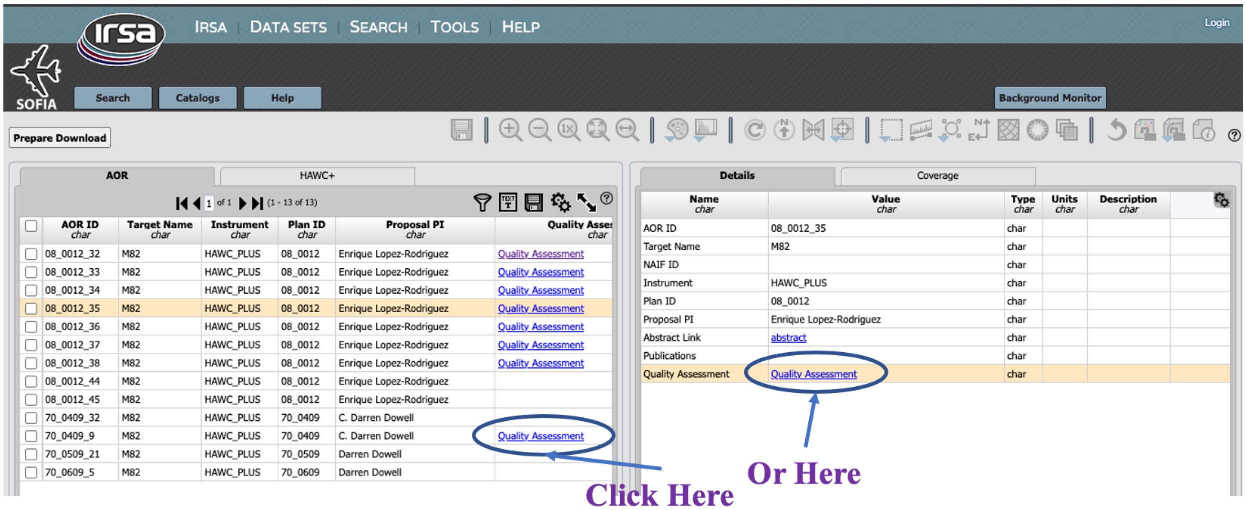
- Review the polarization vectors in DS9.

  Load the final map (19-extension PMP file) using the "open as multiple extension cube" in DS9. From the region tab, click "open" to load the region file that includes the polarization vectors that passed the applied quality cuts, stored as a .reg file in the pipeline output directory.

  Once the region file is loaded on all extensions, you can verify that the concentration of polarization vectors and the peaks of the debiased polarization flux (extension 15: DEBIASED POL FLUX) are in agreement and the quality cuts have been strong enough to filter out unphysical polarization patterns that could appear on the edges of the maps.



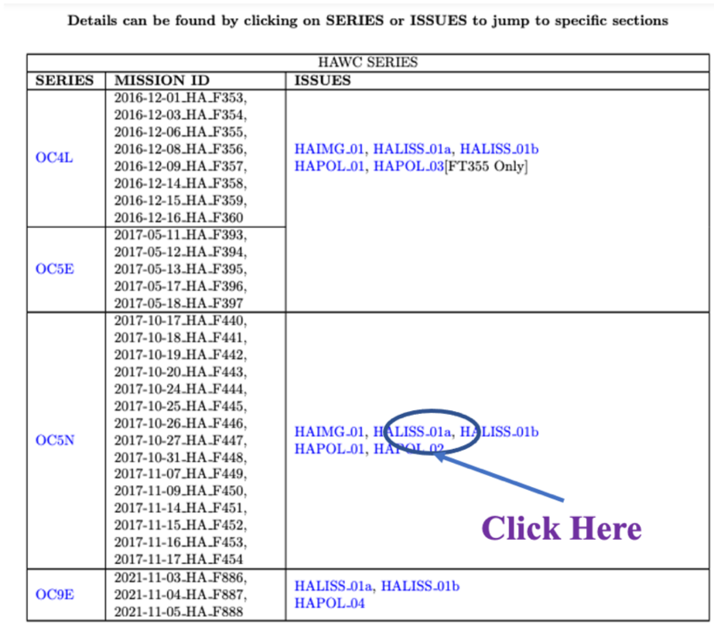
It can also be very useful to check the Quality Assessment information that is supplied by the SOFIA data reduction team. If available, these notes are linked in the AOR tab of the IRSA search results for an observation. For a summary of definition of the keywords used in QA notes, see https://www.sofia.usra.edu/sites/default/files/QASummary.pdf



qa_comments
2020-09-12_HA_F686 (file 169-172): There are some negative pixels in the background regions due to the lack of an emission-free region in the image.

An example of QA notes from a scan-pol observation of the Orion bar (see the Additional Exercise, below).

The data reduction team also keeps a table of the known issues with the data by instrument, for each observing cycle. Through the course of the pipeline development and refinement trajectory, some of the older reported defects and features are not seen in the most recent data products. Likewise, with new shared-risk observing modes, new defects may be discovered. Check the table for the most recent information.



Details can be found by clicking on SERIES or ISSUES to jump to specific sections

HAWC SERIES		
SERIES	**MISSION ID**	**ISSUES**
OC4L	2016-12-01_HA_F353, 2016-12-03_HA_F354, 2016-12-06_HA_F355, 2016-12-08_HA_F356, 2016-12-09_HA_F357, 2016-12-14_HA_F358, 2016-12-15_HA_F359, 2016-12-16_HA_F360	HAIMG_01, HALISS_01a, HALISS_01b HAPOL_01, HAPOL_03[FT355 Only]
OC5E	2017-05-11_HA_F393, 2017-05-12_HA_F394, 2017-05-13_HA_F395, 2017-05-17_HA_F396, 2017-05-18_HA_F397	
OC5N	2017-10-17_HA_F440, 2017-10-18_HA_F441, 2017-10-19_HA_F442, 2017-10-20_HA_F443, 2017-10-24_HA_F444, 2017-10-25_HA_F445, 2017-10-26_HA_F446, 2017-10-27_HA_F447, 2017-10-31_HA_F448, 2017-11-07_HA_F449, 2017-11-09_HA_F450, 2017-11-14_HA_F451, 2017-11-15_HA_F452, 2017-11-16_HA_F453, 2017-11-17_HA_F454	HAIMG_01, HALISS_01a, HALISS_01b HAPOL_01, HAPOL_02
OC9E	2021-11-03_HA_F886, 2021-11-04_HA_F887, 2021-11-05_HA_F888	HALISS_01a, HALISS_01b HAPOL_04

Running the pipeline from the terminal (Automatic Mode Execution)

After you have gained sufficient familiarity with the pipeline processes, you may wish to batch reduce sets of data for more efficient processing, using the command line interface.

The pipeline may be run by directly specifying FITS files on the command line, or with an input manifest, as:

```
> redux_pipe infiles.txt
```

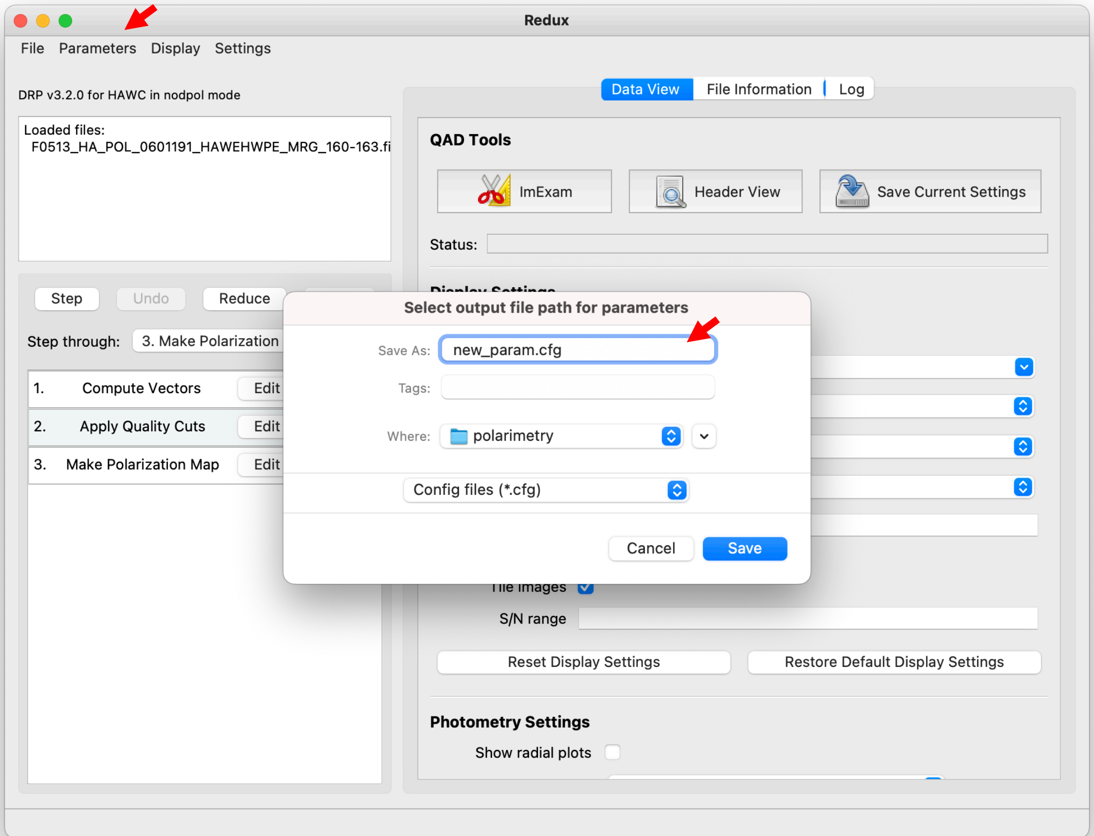
where infiles.txt is a file that contains the full or relative path to the input files. Input files may be raw HAWC+ FITS files or intermediate files, but if multiple files are specified, they must all be of a consistent product type.

For more usage information,

```
> redux_pipe -h
```

shows other command line options to set the output directory, override default parameters, or set the terminal logging level.

To run the pipeline with alternate parameters, it’s easiest to start with a copy of the default values. From the Redux GUI, with your Nod-Pol data loaded, click on the Parameters menu, then select Save Parameters. Name the file new_param.cfg and click Save.



Open the new_param.cfg file in a text editor and edit the 'minisigi' parameter under the [region] section, setting it to 5.0 to show noisier polarization vectors in the final map.

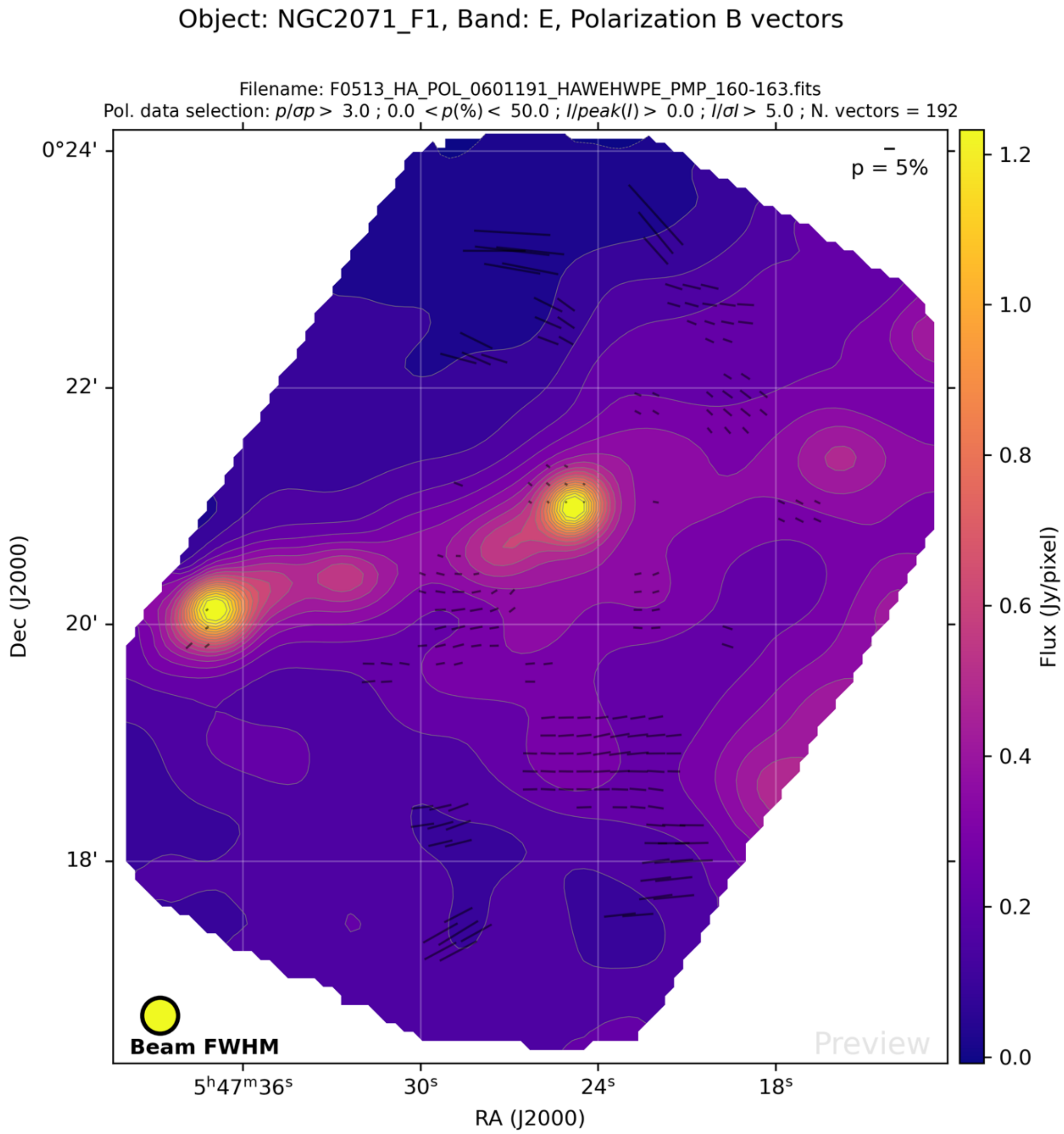
```
# Redux parameters for HAWC instrument in nodpol mode
# Pipeline: HAWC_DRP v3_2_0
[1: polvec]
    save = False
    eff = 0.842, 0.9, 0.939, 0.975, 0.978
[2: region]
    save = False
    skip = 2
    offset = 0, 0
    mini = 0.0
    minp = 0.0
    sigma = 3.0
    minisigi = 5.0
    maxp = 50.0
    scale = True
    length = 10.0
    rotate = True
    debias = True
```

Run the batch pipeline on the intermediate merged data from your pipeline output directory with your new parameters and direct the output to a new folder:

> redux_pipe F0513_HA_POL_0601191_HAWEHWPE_MRG_160-163.fits -c new_param.cfg -o new_output

Open new_output/F0513_HA_POL_0601191_HAWEHWPE_PMP_160-163_polmap.png and compare it to the previous polarization map.

Try editing the configuration file to other values, re-run the redux_pipe command, and compare the output to your previous results.



Final polarization map with minisigi=5.0. Note the unphysically large vectors near the edges of the map.

Additional Exercise: Scan-Pol Mode

In the last few years of HAWC+ operation, polarimetry was offered in a shared risk mode based on continuous scans of the field, rather than chopping and nodding motions. The pipeline steps for Scan-Pol mode perform most of the same procedures as for the Nod-Pol mode, but many of the early artifact detection and removal steps are performed in a single, iterative module rather than being split into separate steps. For more information on the scan mapping procedures, see the scan section of the pipeline User's Manual.

In this exercise, we'll download an observation of the Orion Bar in Band E, taken in Scan-Pol mode, and reduce it through the pipeline steps.

2. Retrieve the data

Click the below link to the data in the IRSA search interface and download all data under the HAWC+ tab, following the same procedure as before:

https://irsa.ipac.caltech.edu/applications/sofia/?api=search&spatialConstraints=allsky&execute=true&processingLevel=LEVEL_0&aorId=08_0209_3

Raw Scan-Pol files come in sets of 4, where each file contains a separate HWP position. Your downloaded zip file should contain 4 POL files, representing one complete set of scans of the Orion Bar (~1.34 GB in size).

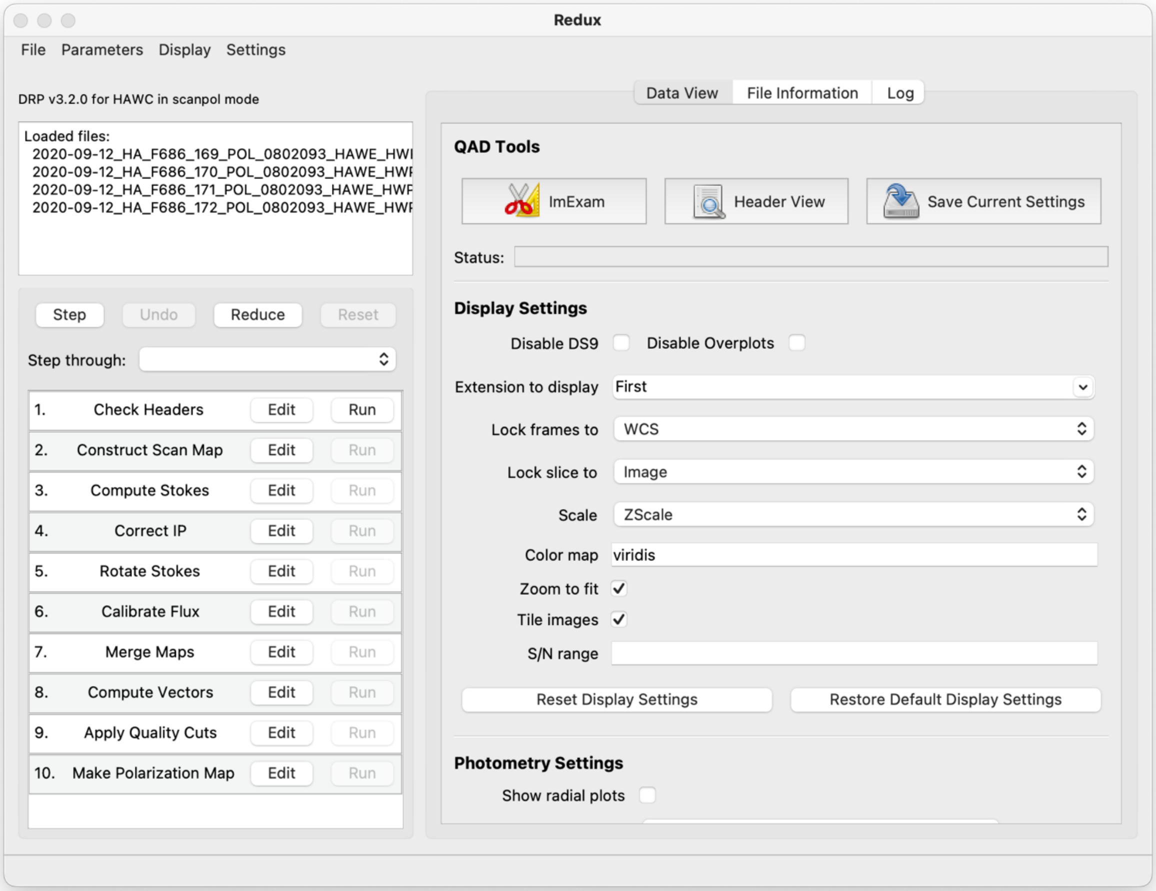
2. Load the data

Navigate to the folder where the data is stored and start the Redux interface:

```
> redux
```

Load in all 4 files with File → Open New Reduction.

There are 10 pipeline modules for Scan-Pol mode. Most of the work happens in the Construct Scan Map step, which iteratively fits and removes correlated noise and detector gains, to leave only a map of the astronomical source structure for each of the R and T arrays at each input HWP position.

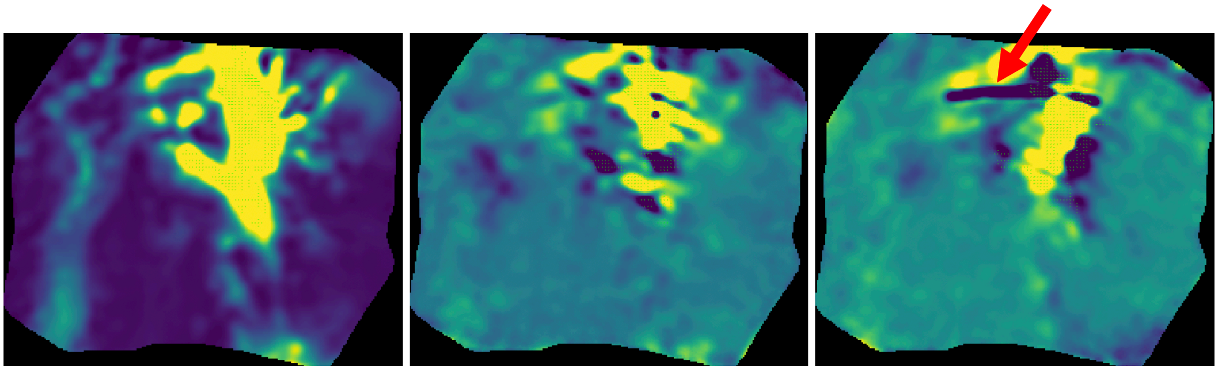


Pipeline modules for Scan-Pol mode.

3. Reduce the data with default parameters

Click Reduce to run through all pipeline steps with default parameter sets. Note that all steps after Compute Stokes are identical to the Nod-Pol process. The final output is a polarization map FITS file (*PMP*.fits) and a preview image (*PMP*.png), in the same format as the Nod-Pol product you have previously examined.

Examine the final map and vectors by loading them into DS9 independently, or by de-selecting the Disable DS9 check box in the Redux interface.

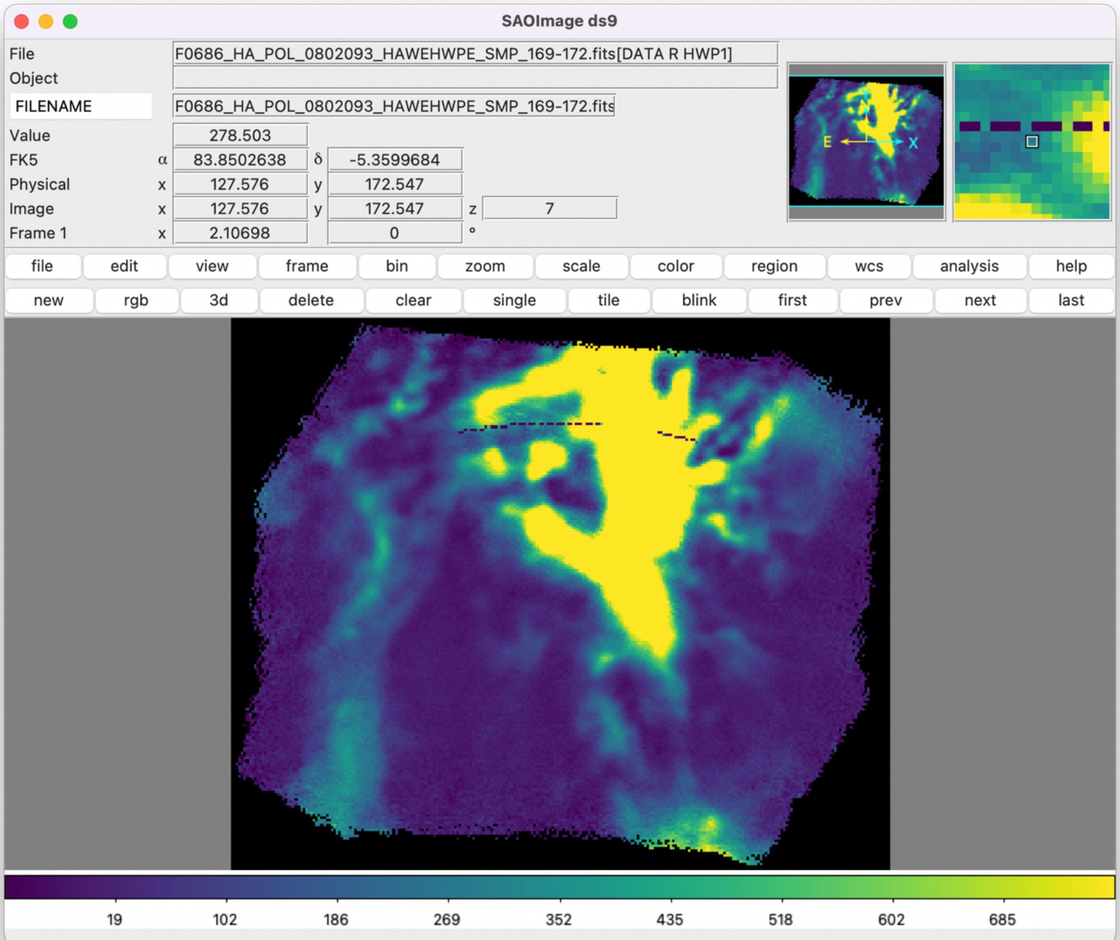


Final product F0686_HA_POL_0802093_HAWEHWPE_PMP_169-172.fits, STOKES I, STOKES Q, and STOKES U extensions. Note the extended unphysical artifact in the Stokes U map.

3. Fix a detector artifact

Note that in the final map produced by the pipeline, above, there is a large extended artifact across the source structure, in the Stokes U map. These artifacts are sometimes produced when there is an extended discontinuous jump in detector flux levels during an observation.

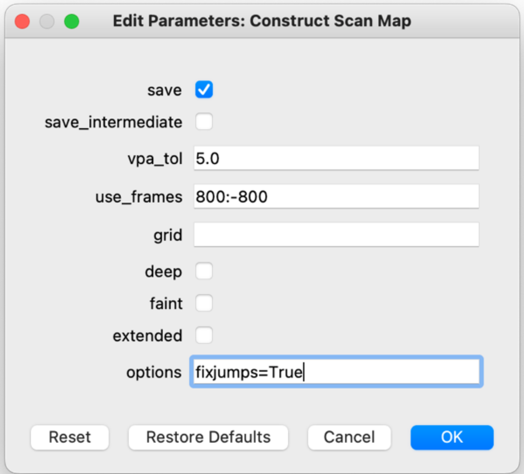
In the final map, the artifact is spread out by the smoothing process in the Merge step. Examining the intermediate scanmap product (SMP file) shows the artifact more clearly as a discrepant detector effect.



Intermediate product F0686_HA_POL_0802093_HAWEHWPE_SMP_169-172.fits, in the DATA R HWP1 extension, shows a large detector jump artifact.

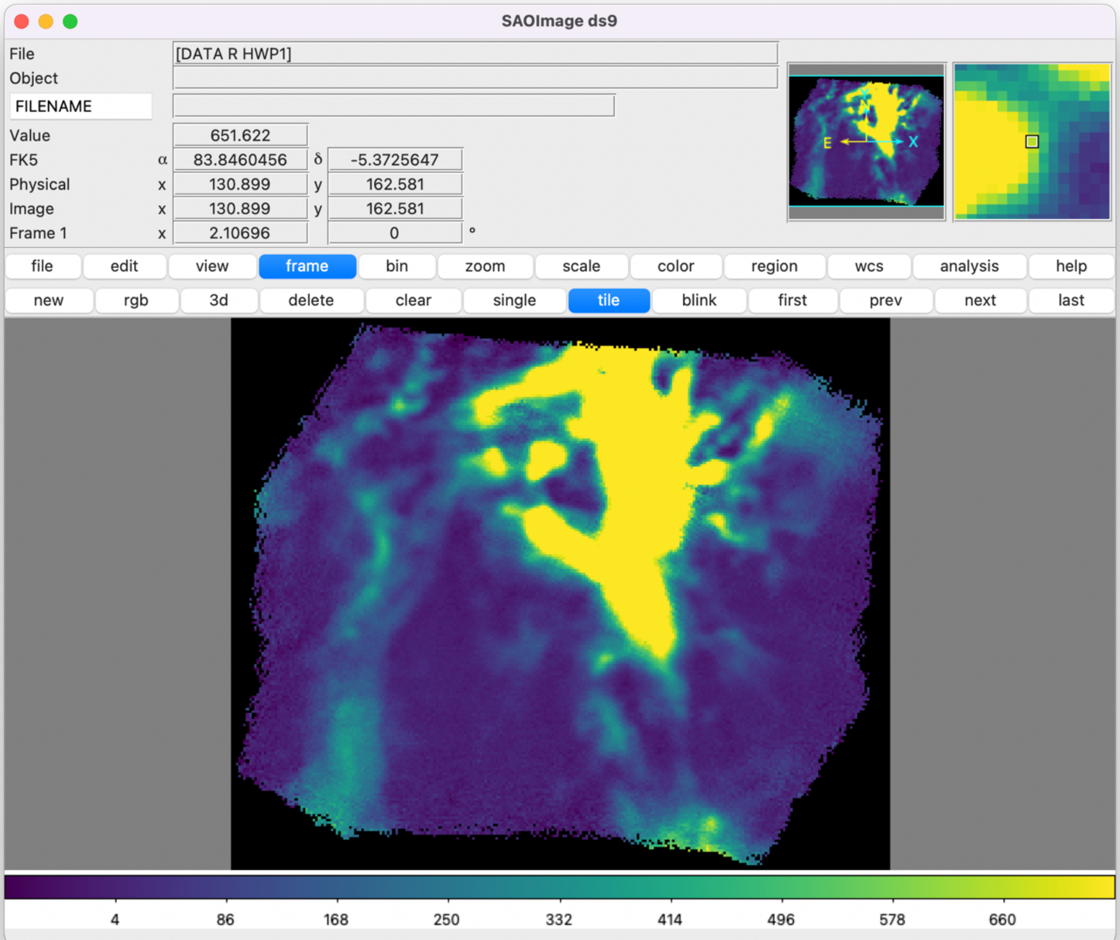
The pipeline can fix these artifacts with a non-default option in the iterative scan map process.

Click Reset to reset the pipeline state, then click Edit next to Construct Scan Map. Next to the options parameter, enter “fixjumps=True”.



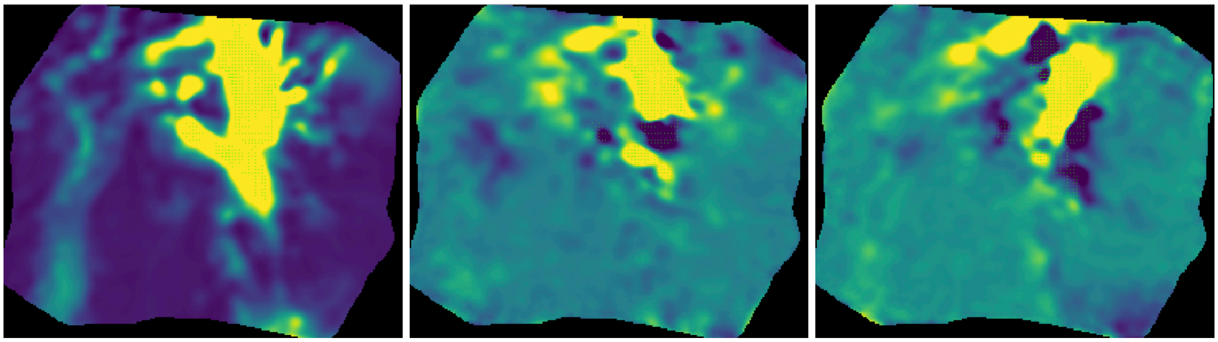
Select Step Through: Construct Scan Map and click Step.

Examine the DATA R HWP 1 extension in the resulting SMP file (extension number 6) and note that it no longer contains the jump artifact.



Intermediate product F0686_HA_POL_0802093_HAWEHWPE_SMP_169-172.fits reduced with fixjumps=True no longer shows an artifact in the DATA R HWP1 extension.

Click Reduce to finish the reduction. The final Stokes and polarized flux maps should now look cleaner too.



Final product F0686_HA_POL_0802093_HAWEHWPE_PMP_169-172.fits, STOKES I, STOKES Q, and STOKES U extensions. Note the artifact is no longer in the Stokes U map.