The STScI STIS Pipeline IV: Combining Repeatobs Data

Philip Hodge, Stefi Baum, and Nelson Zarate
December 6, 1995

ABSTRACT

STIS data can be taken as a series of separate exposures. We describe a two-part program, calstis8a and calstis8b, to combine these data into one image with the equivalent longer exposure time. The raw data will be calibrated by calstis1 as usual. The raw data will also be combined by calstis8a, and the combined data will be calibrated by calstis1. If Doppler shift was significant, calstis8b will be used as well to combine the science image and data quality arrays.

1. Introduction

For both the MAMA and CCD in the STIS, the observer has the option of specifying that a long exposure be broken into a set of two or more subexposures, all with the same pointing, aperture, and grating or filter. The individual exposure times may differ, however. This may be done for time resolution, for example, or to offer some protection against losing all data in case of a telemetry failure. However, it would not be used for cosmic ray rejection; REPEATOBS and CRSPLIT are separate options in a proposal. This ISR describes the calibration program for adding the separate subexposures to make one image equivalent to what would have been obtained if the exposure had not been broken. The input and output file formats are described, as is the role of this program in pipeline processing.

2. Discussion

The format of the STIS FITS files is described in detail in the STIS ISR 95-006. Briefly, data are stored in FITS IMAGE extensions in sets of three, each set consisting of the science data, an error array, and a data quality array. The word "group" will be used in this ISR to refer to such a set of three extensions. The groups of extensions are numbered using keyword EXTVER. The number of these groups in the uncalibrated FITS file, i.e.
the number of repeatobs subexposures, is given by the keyword NRPTEXP. The combined subexposures will written to a FITS file with one group.

During routine pipeline reduction, calstis1 will be run on repeatobs data to calibrate each exposure individually. Calstis8a will be run on the same input data to combine the exposures, and the summed data will then be calibrated by calstis1. In cases where Doppler correction is significant, calstis8b will be run as well. The reason for calstis8b is explained below. The output of calstis8a is a temporary file, and the subsequent output of calstis1 will have suffix “.sfl”. Calstis8b will modify the “.sfl” file in-place.

The combination of repeatobs data is complicated by two factors, the Doppler shift and the propagation of errors. The Doppler shift leads us to combine after calibration, while the computation of error estimates requires us to combine the raw data.

The error estimates for the combined data cannot be computed by merely summing (in quadrature) the error estimates for each group of the calibrated data. This is because the errors in the calibrated data include contributions from both Poisson noise and uncertainties in the reference data, so combining them would improperly reduce the errors contributed by the reference images by the square root of NRPTEXP. So we get the error estimates by running calstis1 on the one-group output of calstis8a, the combined raw data. If the Doppler shift is insignificant, the resulting combined and calibrated science image data and data quality array will also be correct, and we have no further work to do regarding combination of data.

When the Doppler shift is not negligible, calstis1 convolves the reference images with a 1-D smearing function in order to approximately compensate for the Doppler shift during an exposure. This approximation will be really good only if the change in Doppler shift during the exposure is one pixel or less, in which case the Doppler convolution is just a shift. Thus we are more likely to get good calibration, especially flat fielding, by calibrating the separate exposures before combining them. The data quality values are also affected, because the pixel positions in the data quality initialization file need to be convolved with the Doppler smearing function. The purpose of calstis8b, therefore, is to read the multigroup output of calstis1 and combine the science image and data quality arrays. We cannot combine the error array values, but we don’t need to, since we already computed them as described in the previous paragraph.

Calstis8a will read a multigroup FITS file containing uncalibrated data, and it will write a one-group FITS file. If the RPTCORR switch in the input primary header is “PERFORM”, then calstis8a will combine data and will change RPTCORR to “COMPLETE” in the output header. If RPTCORR is not “PERFORM”, calstis8a will simply exit without doing anything. The algorithm for combining the input data is as follows. The values in all science image array (SCI) extensions will be added together, pixel by pixel, with no rejection of data or checking for outliers. The values in all data quality (DQ) extensions will be
bitwise ORed, pixel by pixel. The output error array values will be set to zero, and calstis1 will compute the errors based on a noise model. Calstis8a will update the NEXTEND keyword in the output primary header to three, and it will set the output EXP TIME keyword in the science image extension header to the sum of the EXP TIME values from the input. The output from calstis8a is a temporary file which can be deleted after calstis1 is run with that file as input.

Calstis8b will be run if DOPPCORR is "PERFORM", in which case calstis8b will do the following. The input will be the multigroup FITS file which is the result of running calstis1 on the original data. No new file will be created, but an existing one-group FITS file will be modified in-place; this is the file created by running calstis8a followed by calstis1. The SCI and DQ extensions will be overwritten, but the ERR extension will not be changed. The algorithm for combining science data and data quality values is the same as used by calstis8a. Calstis8b will check the RPTCORR switch in the multigroup input header. RPTCORR in the one-group file will already be set to "COMPLETE" and will not need to be modified again. By overwriting extensions in an existing file, we avoid having an intermediate product from calstis8b.
Figure 1: Flowchart

if RPTCORR = PERFORM

YES

if DOPCORR = PERFORM

YES

sfl* = updated sci and dq extensions