CORONAL DIAGNOSTIC SPECTROMETER
SOHO

CDS SOFTWARE NOTE No. 48

Version 5 27 January 1999

CDS POINTING CALCULATIONS

William Thompson
NASA Goddard Space Flight Center
Code 682.3
Greenbelt, MD 20771

William.T.Thompson.1@gsfc.nasa.gov

1 Introduction

Calculating the pointing of CDS is a somewhat complex process. There are three main areas that affect the pointing:

- The Offset Pointing System (OPS) which controls the large scale pointing properties of CDS.
- The internal mechanisms for the scan mirror and slit, which provide fine scale pointing control for CDS relative to the large scale pointing given by the OPS.
- The pointing of the SOHO spacecraft itself, which acts to modify the pointing given by the various mechanisms.

Each of these areas will be considered in turn. At the end, I'll discuss various software for handling pointing.

2 The Offset Pointing System

CDS is attached to the SOHO spacecraft by six legs, which are free to pivot about their attachment points. Each leg restricts a specific degree of freedom of movement. Two of the legs have actuators attached which allow them to change their length—the other four are fixed length. By changing the length of these two actuators, the pointing of CDS is changed.

The two actuators are attached to CDS at a $\sim 45°$ angle. Changing the length of the left actuator (l) moves the pointing from upper right to lower left, and changing the length of the right actuator (r) moves the pointing from upper left to lower right. To move from left to right, to a first order approximation one would change the difference $l - r$ while keeping the sum $l + r$ constant. The opposite would be true for moving up and down. A more exact model is given by the equations

$$\begin{aligned}
\Delta l &= l - l_0 \\
\Delta r &= r - r_0 \\
\Delta m &= \Delta l - \Delta r \\
\Delta p &= \Delta l + \Delta r \\
x &= -k_x \left(\Delta m + \frac{\Delta m \Delta p}{R} \right) \\
y &= -k_y \left(\Delta p - \frac{\Delta m^2 + \Delta p^2}{2R} \right)
\end{aligned} \qquad (1)$$

where l_0 and r_0 are the actuator positions representing disk center, k_x and k_y are proportionality constants in units of arcseconds/step, and R is a term representing the second order behavior. The results x and y are the pointings of CDS offset from disk center in the E-W and S-N directions respectively, where x increases towards solar west and y increases towards solar north. A number of cross-correlations of CDS images from the FSUN study with cotemporal EIT images have been analyzed, and the best fit to these parameters are in Table 1.

There are two places that the pointing calibration parameters are stored. One place is in the cdhsstate database, which is used by the commanding software, and is maintained as a historical

Table 1: Best fit values to the coefficients in equations 1.

$$l_0 = 2034$$
$$r_0 = 2008$$
$$k_x = 0.953$$
$$k_y = 0.942$$
$$R = 5.69 \times 10^4$$

list of how the instrument was commanded. The other place is in a calibration database, which has more flexibility in allowing for our changing understanding of the true calibration of the OPS mechanism.

2.1 The OPS commanding process

The steps in commanding CDS to a given pointing position are as follows: First of all, the planner selects a given pointing location x, y. This pointing is converted by the command generation software to OPS positions l, r through the approximate inverse of equations 1, and is sent up to the instrument.

There are several uncertainties about the pointing in this process:

- The OPS does not always move to exactly the position it was commanded to. Sometimes it will be 1 or 2 steps off in either l or r, or both. This translates into a few arcseconds uncertainty in the pointing relative to what it was commanded to.
- It appears that the true pointing of CDS is not 100% replicable, even when the OPS moves to exactly the same l and r coordinates. It's likely that there's some play or hysteresis in the mechanisms. The magnitude of this effect is believed to be ~5 arcseconds RMS.
- There is a small uncertainty in the calibration of the parameters in Table 1, at the 1–2 arcsecond level.
- It is also possible that there are additional factors in the conversion process which are not accounted for in equations 1. However, it is believed that this contribution is small.

Given the various contributions, it is a good rule of thumb to not expect to be able to point CDS to better than 10 arcseconds.

2.2 OPS coordinates in FITS files

The OPS coordinates appear in the CDS science telemetry in two places. One place is in the raster header packet that precedes each raster. These values appear in the FITS header as OPS_L and OPS_R, with the comments "OPS L/R position at start of raster". In the quicklook data structure, they can be referenced as `QLDS.HEADER.OPS_L` or `.OPS_R`.

Another place that the OPS values appear is in the exposure header packets that precede each exposure. These are stored as arrays in the FITS binary table, and can be retrieved from the data structure as `QLDS.OPS_LDATA` or `.OPS_RDATA`. The dimensions of the arrays reflect the scanning pattern. However, these arrays are not ordered by time, but by spatial pixel. The relative start times of each exposure are stored in the array `QLDS.DEL_TIMEDATA`, which has the same dimensions.

It seems that these two sets of OPS values—those from the raster header packets, and those from the exposure header packets—do not always agree with each other. They are placed in the telemetry streams via different routes. One possible explanation is that the raster header values may reflect commanded values, while the exposure header values are measured values. In any case, the values stored within the binary table are always more reliable.

When feature tracking is turned on during a raster, then this will be seen as a change in the OPS values in the binary table. The same is true for any other change in the pointing during a raster. However, sometimes one will see an OPS position change by one step during a raster, or change back and forth between two adjacent values, even when there aren't supposed to be any movements of the OPS. This may be caused by the mechanism being at a position halfway between two steps, and switching between rounding up and rounding down.

The pointing derived from the OPS arrays is stored in the FITS header as the parameters INS_X0 and INS_Y0 (`QLDS.HEADER.INS_X0` or `.INS_Y0`). There are two ways in which these values are calculated:

- If feature tracking is turned off, then the OPS values are simply averaged.
- If feature tracking is turned on, then the OPS values are interpolated via a linear fit back to the start of the raster. Thus, the pointing of all the exposures are referenced to the time DATE_OBS, almost as if the data were all taken simultaneously.

3 Internal mechanisms

For a given OPS position, it is possible to observe any point within ± 2 arcminutes of this position by moving the internal scan mirror and slit mechanisms.

3.1 The NIS spectrograph

When observing with the NIS spectrograph, only the mirror mechanism is used, because the NIS slits themselves are 4 arcminutes high. The mirror mechanism can be moved from step position $m = 68$ to 188, with a home position at $m = 128$. The step size is 2.032 arcseconds, and the motion is from right to left. Therefore, the offset from the OPS pointing is

$$\Delta x = 2.032 \times (128 - m) \tag{2}$$

The mirror positions are stored in the FITS binary table in the column MIR_POS, and are accessible from the quicklook structure as `QLDS.MIR_POSDATA`.

The y coordinate depends on the pixel location on the NIS detector. Each NIS pixel is equivalent in height to 1.68 arcseconds. Thus, the 4 arcminute length of the slit corresponds to 143 pixels.

The pointing implied by the OPS and mirror mechanism positions is associated with the center of the slit, and the pixels above or below this are offset in pointing by the factor

$$\Delta y = 1.68 \times (j - j_0) \tag{3}$$

where j is the vertical pixel position on the detector, and j_0 is the position of the center of the slit.

Because the two NIS spectra are slanted on the detector, the position j_0 of slit center is a function of horizontal position i (i.e. wavelength). Applying equation 3 is fairly straightforward when one is dealing with windowed data, because the window positions shift to follow the slant. However, when the full NIS spectrum is read out, the definition of j_0 becomes ambiguous. The convention that has been adopted is to define j_0 as the value that applies at the horizontal center position i_0 of the window. This is what is used when writing out the FITS file.

3.2 The GIS spectrograph

When observing with the GIS, both the scan mirror and slit mechanism are used to offpoint from the OPS pointing, as the GIS is a one-dimensional detector. The slit mechanism moves in steps of 1.016 arcseconds from position $s = 0$, where s can be both positive and negative. Thus, the offset pointing due to the slit can be expressed as

$$\Delta y = 1.016 \times s \tag{4}$$

Like the scan mirror position, the slit positions are stored in the FITS binary table in the column SLIT_POS, and are accessible from the quicklook structure as `QLDS.SLIT_POSDATA`.

4 Spacecraft pointing

The SOHO definitive attitude files can be used to derive the perturbation to the pointing due to the spacecraft. The spacecraft pointing is stored in the FITS header in the parameters SC_X0, SC_Y0, and SC_ROLL. They affect the pointing via the equations

$$\begin{aligned} x' &= x \cos r_{\rm sc} - y \sin r_{\rm sc} + x_{\rm sc} \\ y' &= x \sin r_{\rm sc} + y \cos r_{\rm sc} + y_{\rm sc} \end{aligned} \tag{5}$$

Typically, the spacecraft offsets from disk center $x_{\rm sc}$ and $y_{\rm sc}$ are 0, and the spacecraft roll $r_{\rm sc}$ is in the range $\pm 0.25°$. There appears to be some attitude data with a roll angle of about -2°, but it isn't clear yet whether or not these data are correct.

The parameters XCEN and YCEN (`QLDS.HEADER.XCEN` and `.YCEN`) in the FITS header reflect the spacecraft pointing corrections, as do the pointing parameters associated with the data windows (`QLDS.DETDESC(I_WINDOW).ORIGIN` and `.ROTATION`). Also affected are the `QLDS.INS_XDATA` and `.INS_YDATA` arrays which follow the internal slit and mirror pointing mechanisms. Only those values derived solely from the OPS positions (INS_X0 and INS_Y0, see Section 2) are not influenced by the spacecraft pointing.

5 Software

There are a number of routines for handling pointing information:

GET_CDS_PNTCAL: This routine converts OPS l and r values into solar x and y, taking into account the most up-to-date calibration applicable to the observation date. The calling sequence is

```
IDL> GET_CDS_PNTCAL, DATE_OBS, OPS_L, OPS_R, SOLAR_X, SOLAR_Y
```

UPD_CDS_POINT: Applies the most recent OPS pointing calibration to CDS data. The calling sequence is

```
IDL> QLDS = READCDSFITS(filename)
IDL> UPD_CDS_POINT, QLDS
```

In particular, this routine can take care of an error which was made in some of the FITS files from early in the mission. On April 16, 1996, the pointing of the spacecraft was adjusted by 198 arcseconds. Thus, a different calibration needs be used before that date than for data after that date. Unfortunately, some of the FITS files corresponding to the first few months of the mission were regenerated using the later calibration, and the headers are thus $\sim$198 arcseconds off from what they should be. UPD_CDS_POINT corrects for this.

It also appears that some of the SOHO attitude files may report an incorrect roll angle. UPD_CDS_POINT supports a /NOROLL keyword which tells it to ignore the spacecraft roll.

There is an optional mode which allows the user to specify the pointing to use.

VDS_ROTATE: This routine is used to remove the slant of the spectra from full NIS data, and optionally to remove the line tilt. The default is to rotate about the center of the window, so that j_0 (see Section 3.1) is unchanged. After applying VDS_ROTATE, the y values implied by the window description are applicable to all pixels.

There is an optional /BOTTOM keyword to VDS_ROTATE, which shifts the data down so that the bottom of the window is aligned with the bottom of the slit. When this keyword is used, the y values from the window description are no longer directly applicable.

The syntax of VDS_ROTATE is

```
IDL> VDS_ROTATE, IN_ARRAY, OUT_ARRAY, BAND
```

plus any keywords. For example,

```
IDL> QLDS = READCDSFITS('s7924r00.fits')
IDL> A = GT_WINDATA(QLDS, 0)
IDL> VDS_ROTATE, A, A, 1
```

GT_SOLAR_XY: This routine returns the X and Y coordinates for a CDS data window. The syntax is

```
IDL> GT_SOLAR_XY, QLDS, WINDOWI, SOLAR_X, SOLAR_Y
```

where QLDS is the quicklook data structure, WINDOWI is the window index, and SOLAR_X and SOLAR_Y are the returned arrays. Generally speaking, the returned arrays have the same dimensions as the window itself. Thus, if one uses GT_WINDATA to return the window, then GT_SOLAR_XY returns the X and Y positions for every pixel in the window. If one uses VDS_ROTATE on a data window, then the same VDS_ROTATE can be used on the X and Y arrays.

XCOR_CDS: This routine does a two dimensional cross-correlation between a CDS raster image and an image stored in a FITS file. From this cross-correlation, the relative pointing is determined. This pointing can then be passed to UPD_CDS_POINT.

See also CDS Software Note #41, "Analyzing CDS Data in IDL: An Observers Guide", by S. V. H. Haugan, for a discussion of additional software for retrieving solar positions.