

# A Multipurpose Pointing Analysis Tool

**Colin Cox & Matthew Lallo**

We present software for determining accurate HST attitude from guidestar and velocity data. This permits mapping V2,V3 to RA & DEC. Example applications are discussed, and detailed instructions for use of this software are provided.

## 1.0 Introduction

After an HST observation, there is often the need to reconstruct the pointing history of the telescope, essentially mapping the V2,V3 plane to celestial coordinates. This can be useful in a number of situations such as locating a given target in V2,V3 space. If the target is seen in a small aperture or on a given pixel of a camera, the SI can then be related to the active FGSs, and its focal plane position refined, assuming of course the guidestars and target positions are astrometric, and the FGS to FGS relationship is well known. In other cases, a particular aperture's V2,V3 can be located in RA, DEC, thereby describing where on the sky that aperture was looking. An example of this was a pointing analysis of an FOS scan across Mars, relating spectra to positions on the Martian surface. Other applications include engineering monitoring of jitter at the target and of proper velocity aberration correction at the target position. Trend analyses of motions in the focal plane can also be made more obvious by more frequent pointing reconstructions of this nature. FGS field distortion mapping, among other things, is similarly aided by such a tool, and most recently it found use in OSS during SMOV to perform real-time pointing determinations for the first anxious post-SM observations.

There is a need for a routine, highly accurate, in-house HST attitude determination code; one that includes current FGS alignment and distortion information, accounts for velocity aberration, runs with fine lock or coarse track, 32K or 4K FGS data, and gives good time resolution. To the best of our knowledge, such a program has not existed. During periods when SI to FGS alignment reports were required, we relied on a PASS product to map V2,V3 onto RA, DEC. Though we have found this code accurate and helpful, it also has a number of drawbacks that have presented problems at one time or another in the past. The requestor of the information did not have good control over the guidestar and target coordinates that were used to make the attitude determination. A good understanding of these positions, which are fundamental data in this type of calculation, is critical to interpreting the results. Other issues with the PASS code was that the time resolution seems unnecessarily low; also the code takes target RA and DEC as input, and outputs V2,V3 but does not transform backwards, nor does it provide information about what the V1 axis is doing. In general, the PASS code was somewhat limiting and thoroughly out of our hands.

The authors have developed Fortran code, *findtarget.f*, that utilizes engineering data taken from OMSAT, a data retrieval front-end designed and maintained by SESD. *Findtarget* began life as a set of routines used to analyze and resolve observation pointing and alignment concerns, but has had success tackling a number of issues, like the examples provided above, brought to us by various instrument scientists, SDAs, analysts, and engineers. The demand has been enough to warrant polishing this code, producing documentation, and making it available to whomever might find it useful. We wish to thank those who have shared their expertise with the concepts and who helped us refine the product with valuable comments and suggestions; Pierre Bely, Olivia Lupie, Bruce Toth of SESD, and Ellyne Kinney, Warren Hack, Lisa Sherbert and Roberto Gilmozzi of SOB, and Ed Smith of RSB.

## 2.0 Input

A discussion of the *findtarget's* input must begin with OMSAT, which is managed by SESD and is the front-end used to extract the data required for a pointing determination. The code was designed to accept input in the format of OMSAT output data. For rare instances where real-time analyses are needed Olivia Lupie has written a parser which takes OSS parameter files, and constructs a file with the same format as an OMSAT file, which our code then reads in. We can make this parser available if the need dictates. In most cases however the archived engineering data from OMS is used. If you do not know how to use OMSAT to extract data, please see Bruce Toth in SESD for instructions and an account on the appropriate machine if necessary.

In all instances, *findtarget* requires the high resolution (6 parameter) mode of OMSAT data retrieval, so be certain to specify such. The primary input file to *findtarget* is what we call the FGS file, which contains star selector encoder angles, and fine error signal PMT counts if the data is fine-lock. *Findtarget* will convert this information to the V2,V3 positions of the guidestars, then fit these coordinates to the RA, DEC of the stars to ultimately determine telescope attitude. There are two input cases, that of coarse track and fine lock. If the data is in coarse track, one extracts from OMSAT, for the active FGSs and time period desired, the Star Position Coordinate parameters, FxSPCR1, and FxSPCR2 from the FGS/FGE subsystems option, where *x* is the FGS number. These two sets of values get extracted for the two FGSs resulting in 4 columns of data, plus time tags that OMSAT embeds, and that *findtarget* uses. The file should look like this:

```

OMSATING.OMS CCCC----- START:179:18:00 END:179:18:20 MAX_REC: 1478 NUM_PAR: 4 ITERM: 2
  Spacecraft Time          F2SPCR1      F2SPCR2      F3SPCR1      Parameters
                          F355          F358          F655          F3SPCR2
                          F658
92.179:18:00:00.126      290348.90625  552599.81250  193032.40625  424962.96875  48800.75000
92.179:18:00:00.626      290346.43750  552603.56250  193032.40625  424962.96875  48800.75001
92.179:18:00:01.126      290346.43750  552603.56250  193031.17188  424960.50000  48800.75001
:
:
:

```

For the fine lock case, again choose the FGS/FGE subsystems option, but this time extract for each FGS, two Star Selector Positions (FxASSP, FxBSSP) and four PMT counts (*x* and *y* for PMTs A and B, FxXFCPA, FxXFCPB, FxYFCPA, FxYFCPB). Since only 6 high resolution parameters can be written to an OMS file, two files must be created for fine lock, one for each FGS. An example of one of the files follows:

```

OMSATING.OMS IIIIRR----- START:099:14:40 END:099:15:05 MAX_REC: 1001 NUM_PAR: 6 ITERM: 2          23-NOV-93 14:38:51 H
Spacecraft Time          F1XFPCA      F1XFPCB      F1YFPCA      Parameters  F1ASSP      F1BSSP
                          F001          F003          F005          F1YFPCB     F007          F009          F012
93.099:14:40:00.195     77.00000     66.00000     82.00000     99.00000     317336.46875 536291.56250 49086.61111
93.099:14:40:00.695     90.00000     73.00000     70.00000     91.00000     317333.37500 536292.18750 49086.61112
93.099:14:40:02.195     86.00000     75.00000     102.00000    86.00000     317327.18750 536298.93750 49086.61114
.
.
.

```

The final file needed from OMSAT is the velocities file, containing the net spacecraft velocities bracketing the time of the FGS data extraction. The PCS Output subsystem option is chosen and the three net velocity components, parameters QDVNETV0, QDVNETV1, and QDVNETV2 are extracted. Below is a sample of this file's format:

```

OMSATING.OMS RRR----- START:099:14:40 END:099:15:05 MAX_REC: 26 NUM_PAR: 3 ITERM: 2
Spacecraft Time          QDVNETV0      QDVNETV1      QDVNETV2      Parameters
                          Q680          Q681          Q682
                          m/s          m/s          m/s
93.099:14:40:00.195     2453.48462    -26287.83594   -8812.82813   49086.61111
93.099:14:40:02.195     2557.88306    -26737.69531   -8640.86719   49086.61114
93.099:14:41:02.195     2692.27368    -27183.67969   -8479.73438   49086.61183
.
.
.

```

The FGS data and the velocities are the only files for which OMSAT is needed. Before moving on, mention should be made of telemetry rates. With 4K engineering data, there will typically be 2 measurement a second, resulting in a file with 0.5 second intervals between time-stamps from line to line. However, if the observation had 32K telemetry, the OMSAT output becomes more complex and lines of data are produced much more frequently than the values are actually updated. This results in needlessly large files. A filtering program, called *fnfilter* will be made available. This allows the user to greatly reduce the size of the FGS data file, and hence speed processing by *findtarget* without losing accuracy. A further inconvenience of 32K data is that currently, OMS net velocities are unavailable for this mode, and therefore the velocity aberration correction is not done for the high rate data. If the need dictates, velocities can be obtained from MOSS and parsed to be read by *findtarget*. Omitting the velocity correction subjects the results only to the differential aberration effects, which will never exceed 40 milli-arcseconds.

The user must create a third file, called the stars file. It contains the celestial coordinates of the two guidestars and either V2,V3 points if RA, DEC results are desired, or celestial coordinates of the target stars, if the focal plane location of those positions is sought. The first line must provide an indication to *findtarget* of which way to transform, and is given by the word V2 or RA, in any location or case. Two examples will clarify:

Ex.1

```

Find RA & DEC
114.0333333 24.2032222 primary g.s.
113.9714167 23.8431111 other g.s.
-128.929    158.662    aperture_A
-129.349    161.614    Ap_B

```

Ex. 2

```

v2,v3
13.15305 85.00493 FGS 3
11.66259 85.39251 FGS 1
11.714475 85.242485 targ1
11.718264 85.243021 targ2
11.724688 85.243411 targ3

```

The format of the star file is relatively free. The first line is only required to have the letters 'ra' or 'v2' (but not both) somewhere, but the line may contain anything else. Upper or lower case is accepted. The next two lines are only required to have a free format RA and DEC in decimal degrees, though one may follow them with any sort of identification as in the examples. The subsequent lines are free format V2,V3s in arcseconds or RA,DECs in decimal degrees. These are the target lines, and must be followed by some

identification that does not contain spaces or special characters since this label will be used by the program to name its output files.

There is one more input file, MINISCHF.DAT. This file is usually constant and is not specified each running. It contains the FGS-FGS alignment matrices, FGS field distortions, k factors, and other parameters relating to the calibration of the FGSs. It was created from the PDB's SCHF.DAT file, but is a trimmed-down version. A current copy of this file is in /tib/u1/lallo/public and can be mailed to anyone not able to copy it. If updates are made to the FGS calibrations, the users of *findtarget* will be informed and a new MINISCHF.DAT file made available. *Findtarget* automatically looks for a file called MINISCHF.DAT in the same directory where the program runs, and if it does not find it will produce a screen message asking for it.

### 3.0 Running *findtarget*

The source code, *findtarget.f*, contains comments at the beginning, briefly summarizing the input and giving build instructions as well. To compile and run with UNIX's *f77*, simply type:

```
% f77 -e -misalign findtarget.f -o findtarget
% findtarget
```

This will create and run an executable called *findtarget*. Should cruel fate befall you that you must compile and run this on a VAX, type:

```
$ FORT/EXTEND_SOURCE FINDTARGET.FOR
$ LINK FINDTARGET
$ RUN FINDTARGET
```

Of course after compiling you may simply run the executable in the future.

If the appropriate data has already been extracted for the time period of interest, and the star coordinate file has been made, then the program can be run. Below is a screen dump of two sample runs. In the first case, the data was in coarse track and the star file told *findtarget* to find RA, DEC from V2,V3s provided. The second running was with fine lock data, and with the code being told to output V2,V3s from the input RA, DEC

Ex. 1 (coarse track, with velocity correction, ra/dec output, 2 targets)

```
squonk.stsci.edu% findtarget
  Is the data in fine lock?
no
  Guide star file
coarsefgs
  FGS A = 2   FGS B = 3
  Give file name containing star positions
coarsestars
  Do you wish to perform velocity aberration correction?
yes
  Give the velocity file
coarsevel
  Results written to the file apertureA.radec
  Results written to the file apertureB.radec
```

Ex. 2 (fine lock, no velocity correction, v2/v3 output, 3 targets)

```
squonk.stsci.edu% findtarget
  Is the data in fine lock?
yes
  First FGS fine error signal file
fineprimary
  Second FGS fine error signal file
finesecondary
  Give file name containing star positions
finestars
  Do you wish to perform velocity aberration correction?
no
  Results written to the file star1.v2v3
  Results written to the file star2.v2v3
  Results written to the file star3.v2v3
```

When running *findtarget*, care must be taken to place the guidestar positions in the stars file in the correct order. For coarse track, this will be in the order corresponding to the FGS columns in the coarse track file. For fine lock it is the order in which the fine lock FGS files are chosen as input. There is no implication that the primary guidestar is first. The program will not detect a reversal of star order but will calculate as if the telescope was rotated 180° about the midpoint of the guidestars. In some cases this will place the target position outside the field of view, triggering an error message.

When running *findtarget* on large amounts of data, there will be a delay after the FGS file(s) are input, while the program transforms FGS data into V2,V3s. Once finished, it then prompts the user for the stars file and velocity file if needed. At this point it goes off and continues processing to produce the output. It thus runs in two stages so the user should remember that input will be required at two distinct times.

## 4.0 Output

*Findtarget* has 2 types of output product, the main output of interest is a .v2v3 or .radec file for each target in the stars file. A line of output is produced for every line of input in the FGS file(s). An intermediate file is also produced, called GUIDESTARS.V23, containing V2,V3 positions of the guidestars in both FGSs calculated from the raw FGS data. This is then read by the program to perform the fit. The GUIDESTARS.V23 file is not deleted and may or may not be useful to the user. The .v2v3 or .radec file for the targets however, contain the target transformations. A sample of the output file is below:

Time (hours)	Telescope Attitude (V1 axis)			Target		Guidestar sep (o-c)	
	RA	DEC	Roll	V2	V3	(diff in arcsec)	year/day/h/m/s
14.66672000	114.106698	24.006534	-80.526834	-26.711	352.363	0.274	93.099:14:40:00.195
14.66686000	114.106696	24.006536	-80.526622	-26.491	352.385	0.280	93.099:14:40:00.695
14.66728000	114.106695	24.006538	-80.526750	-26.216	352.364	0.278	93.099:14:40:02.195

The time is given in decimal hours in the first column, and hours, minutes, seconds in the last column. The decimal hours time-string is particularly useful for time series plots. Columns 2, 3, and 4 are the Heliocentric RA and DEC of the V1 axis and ROLL of the telescope. Columns 5 and 6 are the transformation of whatever positions were given for the targets in the stars file, into the other coordinate system, based on the telescope's attitude, and are usually the main columns of interest. Column 7 requires some explanation; it is essentially an observed-minus-calculated measurement. It is the difference, in arcseconds,

between the separation of the guidestars' celestial coordinates, compared with the separation of the guidestars' computed V2,V3 positions. This indicates how accurately both the celestial positions of the guidestars, and the alignment and distortions of the FGSs are known. Unfortunately, the two effects cannot be further disentangled from this alone. The value serves as an indicator of the uncertainty of the target transformation, although the relationship of the guidestar separation error on target error depends on the guidestar-target geometry. Spuriously high values of this measurement will occur if *findtarget* is applied to time periods when the telescope is slewing.

Finally, it should be kept in mind that, assuming all other related calibrations are accurate, an attitude determination of this sort is only as good as the guidestars' and targets' celestial coordinates as provided in the stars file. Although the operational system may have used guidestar catalog coordinates for the observation (which have uncertainties of order 1 arcsecond), the post-observation pointing reconstruction will always benefit from use of the most accurate star positions available. However, using the GSC coordinates for the guidestars, and the target coordinate provided in the SMS, should reproduce the planned pointing. Any deviation from this result indicates calibration errors such as inaccurate FGS-FGS alignment matrices.

## 5.0 Appendix

### 5.1 error messages

Discussed here are error messages that may be encountered while running *findtarget*, and a description of program behavior, ordered as would be encountered during runtime:

- If the response to 'is the data in fine lock?' or 'do you wish to perform velocity aberration correction?' does not begin with a 'Y' or 'y', the code interprets that as no.

```
THE MINISCHF.DAT FILE WAS NOT FOUND. IT IS USED BY THE CODE TO OBTAIN FGS  
CALIBRATION INFORMATION. A COPY OF THIS FILE CAN BE OBTAINED UNDER /tib/  
ul/lallo/public. PLEASE PROVIDE AND THEN RUN FINDTARGET AGAIN.
```

- This message is self-explanatory. The only issue is that if changes are made to the FGS matrices or distortion models, the MINISCHF.DAT file must change to reflect the current knowledge. The authors will be aware of such changes and will provide a new file.

```
THERE IS NO FILE NAMED XXX PLEASE RE-ENTER.
```

- Self-explanatory. Indicates that the particular file was not found *in the directory from which the program is running*.

```
BAD DATA AT LINE XXX IN examplefgs ...IGNORED
```

- *Findtarget* has found unintelligible data in a line of an FGS file. It will skip to the next line and resume processing. In the fine lock case of two FGS files, it will skip down in the other file also to remain synchronized in time. It is not unusual to encounter, in very long OMS data files, extraneous characters or format inconsistencies, and the program deals with these irregularities fairly gracefully.

ERROR IN DATE STRING AT LINE XXX

- *Findtarget* has found a nonsensical time-string, and will skip this line.

THE FIRST LINE OF THE STAR FILE MUST CONTAIN EITHER AN UPPER OR LOWER CASE RA OR V2 TO INDICATE WHICH WAY TO TRANSFORM RESULT. PLEASE EDIT THAT FILE AND RERUN.

- If you already have indicated the mode in the first line, but are still getting this message, be sure that the case is consistent ('RA' or 'ra' rather than 'rA'). Avoid mentioning both coordinate-system keywords in the same line.

ERROR READING VELOCITY FILE - INTERPOLATING TO XXX.

- XXX is a timestring, allowing the user to find the location of the bad data. The code will just interpolate using the next good line of data.

THE TARGET IS OUTSIDE HST FOV

- This message might arise if the order of the guidestar coordinates in the stars file does not match the order the FGS data was input, or if erroneous data of the right format was encountered. The celestial coordinates of the guidestars and the target may also be suspect.

REACHED END OF VELOCITY FILE.

- This message is produced if the time of the velocity file's last readable line is earlier than one or more data lines in the FGS file(s). This would occur if the velocity file time period does not bracket that of the FGS data file.

## 5.2 *algorithm details*

The calculation proceeds in three steps, converting FGS measurements to V2, V3 values, determining the telescope attitude, and calculating target positions in the field of view. Converting the FGS lever arm measurements into V2 V3 values is performed by the subroutine *fgs\_to\_veh* which has a long history. According to the comments in its header, the original algorithms are from Perkin-Elmer and were implemented by CSC in the PASS system. The routine was modified by Lattanzi and Bucciarelli to incorporate in OMS. There is no indication that any algorithmic changes were made. The routine refers to the FGS-to-FGS matrices and the optical distortion model of the FGSs, the data for which are maintained in the SCHF.DAT file in the Project Data Base.

By matching the RA and DEC's of the guide star pair against the calculated V2 and V3 the telescope attitude can be determined. The attitude matrix  $M(\alpha, \delta, r)$  is a function of the RA,  $\alpha$ , the DEC,  $\delta$  of the V1 axis and the roll,  $r$  about that axis. A unit vector  $\mathbf{u}$  is constructed from the RA and DEC of each guide star. If the telescope velocity is supplied, we apply the velocity aberration correction to transform into the telescope's velocity frame. For each star, another unit vector  $\mathbf{v}$  is derived from the V2 V3 values. For the correct attitude,  $M(\alpha, \delta, r)\mathbf{u}$  gives the unit vector in the telescope's frame and should be identical to the vector  $\mathbf{v}$ . The two stars provide four measurements to determine the three unknowns. An extra condition is provided by the restriction that the angular separation between the guide stars should be the same when determined from the RAs and DEC's as when calculated from the V2s and V3s. Due to measurement errors, the condition is not exactly met and a

solution is found by least squares estimation. We choose  $\alpha, \delta$ , and  $r$  to minimize the value of  $\Sigma |\mathbf{u}-\mathbf{v}|^2$ . The minimization algorithm is the Polak-Ribiere conjugate gradient method, given as the routine FRPRMN in Numerical Recipes.

Once the telescope attitude is known we may calculate the V2, V3 of a series of targets of known RA and DEC or the RA and DEC's corresponding to V2, V3 positions. In the former case, we obtain the unit vector in celestial coordinates, apply the velocity aberration correction if specified, and transform to telescope coordinates. The V2 V3 to RA and DEC conversion proceeds by generating a unit vector in telescope coordinates, converting to celestial coordinates using the inverse of  $M$  and finally applying the velocity aberration correction in reverse to obtain heliocentric coordinates.



# DISTRIBUTION

## **SPD**

Blades  
Macchetto

## **SIB**

Baggett  
Biretta  
Burrows  
Clampin  
Evans  
Gilliland  
Greenfield  
Hack  
Hulbert  
Keyes  
Kinney, A.  
Koratkar  
Krist  
MacKenty  
Nota  
Paresce  
Ritchie  
Soderblom  
Sparks  
Taylor  
Sherbert

## **SOB**

Bohlin  
Colina  
Cox  
Gilmozzi  
Hasan  
Holfeltz  
Kinney, .E  
Taff  
Tullos

## **SESD**

Bely  
Doxsey  
Lupie  
Toth  
Wheeler  
Bielefeld

## **RSB**

Golombek  
Smith

## **SPB**

Noll  
Lubenow  
Weaver