orbit

April 16, 2023

Abstract

Task to create a FITS timeseries "HK-like" file from the ODF Spacecraft Predicted or Reconstructed (*ROS.ASC) Orbit File.

1 Instruments/Modes

Instrument	Mode
EPIC	ALL
RGS	ALL
OM	ALL

2 Use

pipeline processing	yes
interactive analysis	yes

3 Description

This task makes position information from the XMM ODF available as a FITS format product file. It has been determined that this product would have value for constructing Good Time Intervals (using **tabgtigen**) based on orbit information, for providing XMM positions in the same coordinate system (GSE) as the ACE and WIND satellites for Trend Data construction (and SWCX analysis), and for possibly creating barycentric correction tables. A FITS format file conforming to both established Orbit file conventions and NASA OGIP FITS standards are more useful than the ASCII Orbit file provided in the XMM ODFs.

This task will utilize the OAL_getPosition routine to return the x-y-z components of the position vector in km as well as the x-y-z components of the corresponding velocity vector in km/s, both with respect to the reference Kepler orbit. Both sets of values will be written into a standard OGIP-compliant FITS binary extension table with a TIME column value, derived from a start, stop and sampling interval time, each specified by the user. The default is to sample the entire input file with a timestep interval of 60.0 seconds. The OAL interpolates the positions for times outside of an observation, and the user is warned

that if such times are outside an ORBIT of a given ODF, the interpolation may fail or take very long periods of time to calculate.

The GEI to GSE coordinate conversion utilizes IDL routines converted to Fortran90 that originated in the Hapgood (1992) [2] paper.

Expanding the use of orbit, and taking advantage or the extrapolation routines in the OAL, one can select times that are far outside the range of a given OBSID. However, because of time and memory constraints, the selection is limited to 172800 seconds before the first time in the current OBSID and 172800 seconds after the last time in that OBSID. This solves the issue of taking a very long time or crashing with a seg fault if the user selects times too far away from the current obsid.

The ODF Orbit file is described in [1]. The Periodic Housekeeping data is described in [2].

4 Parameters

This section documents the parameters recognized by this task (if any).

Parameter	Mand	Type	Default	Constraints
outfile	yes	string	none	valid file name

Output orbital HK timeseries filename (FITS).

selecttimes	no	boolean	no	yes—no

User inputs starttime and endtimes (T), or use (default) whole file (F)?

starttime	no	real—string	none	>0.0 s

Start time in seconds since XMM reference time.

endtime	no	real—string	none	>0.0 s

End time in seconds since XMM reference time.

timestep	no	real	60.0	>0.0 s

Bin size (in sec) of output timeseries.

useobsid	no	boolean	yes	yes—no

Constrain output to OBSID or use whole orbit?

Note: Users can enter starttimes and stoptimes 172800s -/+ the OBSID's start and end.

5 Errors

This section documents warnings and errors generated by this task (if any). Note that warnings and errors can also be generated in the SAS infrastructure libraries, in which case they would not be documented here. Refer to the index of all errors and warnings available in the HTML version of the SAS documentation.

TotalTimeLEzero *(error)*
End - Start less than or equal to 0.

StarttimeGTEndtime *(error)*
Starttime entered greater than Endtime entered

StartGTorbit *(warning)*
Start time entered more than 2d after orbit end
corrective action: Default to OBSID start

EndGTorbit *(warning)*
End time entered more than 2d after orbit end
corrective action: Default to OBSID end

StartLTorbit *(warning)*
Start time entered more than 2d before orbit start
corrective action: Default to OBSID start

EndLTorbit *(warning)*
End time entered more than 2d before orbit start
corrective action: Default to OBSID end

StartZero *(warning)*
Start = 0, selecttimes=yes
corrective action: Default to OBSID start

EndZero *(warning)*
End = 0, selecttimes=yes
corrective action: Default to OBSID end

badTimestep *(warning)*
Timestep parameter entered less than 0 or greater than time in file
corrective action: Default to 60 sec

6 Input Files

1. None. SAS_ODF must be properly set.

7 Output Files

1. FITS file with single binary extension containing twelve) columns: TIME in seconds since MJDREF, x-y-z GEI position vectors in km,x-y-z GSE position vectors in km, (GEI) x-y-z velocity vectors in km/s, sun angle and RAM angle.

8 Algorithm

```
subroutine orbit
 read_parameters    # get parameters from param file
 OAL_odfInfo        # get StartTime, EndTime of file

 open_outfile       # open FITS output file
 addTable           # create new table
 foreach column     # create 10 new columns: time, x, y, z, Vx, Vy, Vz
  addColumn
 next

 for start_time to stop_time, step sampling_interval

  OAL_getPosition   # return position and velocity vectors from ODF

  foreach column    # time, x, y, z, Vx, Vy, Vz
   fill_column_temp_pointer
   if (column_temp_pointer(i) not defined)
    column_temp_pointer(i) = HUGE
   end if
   convert_GEI_to_GSE(time,gei_x,gei_y,gei_z,gse_x,gse_y,gse_z)
 ! gei_x,y,z : celestial x,y,z of XMM in km.
 ! gse_x,y,z : geo solar ecliptic x,y,z of XMM in km.
 ! lambda = ecliptic long of sun  (radians)
 !    eta = obliquity of ecliptic (radians)
 ! The Sun's ecliptic longitude (lambda0) can be calculated using the
 ! series of formulae:
 !
 !  meanAnom = 357.528 + 35999.050T0
 !  Lambda = 280.460 + 36000.772T0
 !  lambda0 = Lambda + (1.915 * sinM) + 0.020 sin2M
 !
 ! where T0 is the time in Julian centuries from 12:00 UT on 1 January 2000
 ! to the midnight Universal Time (UT) preceding the time of interest and
 ! H is the time in hours since that preceding UT midnight. Formulae
 ! derived from the Almanac for Computers. In the intermediate formulae,
 ! meanAnom is the Sun's mean anomaly and Lambda its mean longitude.
 !
 ! M = 357.528+35999.050*Tzero+0.04107*UT; Sun's mean anomaly
 ! L = 280.460+36000.772*Tzero+0.04107*UT; Sun's mean longitude

  next

 next

 OAL_proposalInfo   # get basic info to write as keywords
 write_attributes_to_outfile     # add proposal info as keywords
 release Table      # close FITS extension
 release Set        # close FITS file
 OAL_releaseMemory  # close ODF files
 close_outfile
end subroutine orbit
```

9 Comments

- This task simply converts some of the available Orbit parameters from their current ODF ascii format (as a Spacecraft Reconstructed Orbit File). It is assumed these positional and velocity parameters can be better accessed in FITS for constructing possible GTI's.
- This task also converts the Geocentric Equatorial Inertial (GEI) in km to Geosynchronous Solar Ecliptic (GSE) in km to facilitate the NASA/GSFC Trend Data construction.
- It is not clear the usefulness of the orbit file for barycentric correction but it is clear this file should be presented in a format that maximizes using preexisting software to calculate such values FROM this file.

The ODF directory is set via the environment variable SAS_ODF. Full use of the OAL calls is required to determine the state and availability of the particular ODF desired. These calls are described in the XMM ODF Access Layer library **oal**. Other OAL calls can be exploited to retrieve information useful for the output Orbit file as keywords.

The position and velocity vectors returned by OAL_getPosition are extra/interpolated by that subroutine.

It does not appear that any information on the error of the orbit measurements will be available to the PPS. This deficiency should be addressed. As the access subroutine returns orbit parameters for any time specified, the user will need to input the time interval for sampling the orbit parameters.

The value HUGE (ieee NaN) is substituted for any column not correctly retrieved or inter/extrapolated by the OAL.

10 Developers notes

10.1 CAL usage

None forseen.

10.2 OAL usage

OAL_odfInfo
OAL_proposalInfo
OAL_toTimeTag
OAL_getPosition
OAL_addCommonAttributes

11 Future developments

Other coordinate system transformations may be added. Other time systems (e.g., YYYY-MM-DDTHH:MM:SS format) may be available for the start and end times when selecttimes=yes.

References

[1] The XMM ODF Access Layer Library, September 30, 1998.

[2] Hapgood, M, Space Physics Coordinate Transformations: A User Guide, Planet. Space Sci, 40, 5, 1992.

References

[1] ESA. XMM Interface Control Document: Observation and Slew Data Files (XSCS to SSC) (SciSIM to SOCSIM). Technical Report XMM-SOC-ICD-0004-SSD Issue 2.5, ESA/SSD, June 2000. Found at the URL: `ftp://astro.estec.esa.nl/pub/XMM/documents/odf_icd.ps.gz`.

[2] K. Galloway. XMM technical note: Periodic housekeeping telemetry definition. Technical Report XMM-SOC-TN-0040-SSD 0.2, ESA/SSD, August 5 1999.