



stspproc

January 12, 2017

Abstract

This task produces the RTX files from Raw TM files. The RTX basically contain the time correlation data in the form of OBT-ERT pairs, as extracted from the raw STSP packets, and fitted by a general Levenberg-Marquard algorithm. The RTX files are known as TCX when found incorporated to the ODFs.

1 Description

This task is intended to read XMM Raw TM files, process the STSP (Spacecraft Time Source Packets) packets and produce the RTX files from such an information.

2 Use

The most simple call is,

```
stspproc -prewindow -postwindow -revolution= -useS2K -version -strict -markbad -cachedir
```

This simple call is usually wrapped by calling dedicated shell script:

```
stsp_runrev.sh rev
```

Or even better, by a master caller script,

```
stsp_runtcx.sh (rev)
```

pipeline processing	no
interactive analysis	no
ODF TCX production	yes

3 Description

STSP packets contain the OBT timestamp. By getting the ERT of such packets, a fitting process can take place for OBT-ERT pairs.



stspproc will open the Raw TM files within a given time span, and read the data searching from VC0 frames, STSP packets.

This is achieved by,

1. - Generating a DU Extractor, with defined start-end boundaries.
2. - Generating a STSP Processor, with defined start-end boundaries.
3. - Generating a NCTRS Processor, hanging from STSP Processor, specialised in VC0 frames.
4. - Generating a Packet Processor, hanging from the STSP Processor.

And then,

1. - A loop is created, for the raw files required sequence reading and processing.
2. - The RTX files are created, based in the STSP packet lists in memory. (pairs of OBT-ERT)
3. - The STSP prior info is corrected (correct STSP process), by a fitting process, spike removal, etc...

See the algorithm section for a detailed description of the issue.

4 Parameters

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

Parameter	Mand	Type	Default	Constraints
-----------	------	------	---------	-------------

revolution	yes	type	void	none
-------------------	-----	------	------	------

Revolution to process

version	no	int	void	none
----------------	----	-----	------	------

Version of RTX file to create (VERSION keyword in FITS)

useS2K	no	bool	no	no
---------------	----	------	----	----

Use S2k data files

strict	no	bool	yes	none
---------------	----	------	-----	------

Abort loop processing when DU sizes errors are found

prewindow	no	real	120000.0	none
------------------	----	------	----------	------

Time margin on before start of revolution to process

postwindow	no	real	80000.0	none
-------------------	----	------	---------	------

Time margin on before after of revolution to process

maxdelta	no	real	5e-5	none
-----------------	----	------	------	------

Maximum acceptable Handover jump

maxresid	no	real	0.05	none
-----------------	----	------	------	------



Maximum data residual

maxtimejump	no	real	1000.0	none
--------------------	----	------	--------	------

Maximum allowed jump in time over handover

symthres	no	real	0.2	none
-----------------	----	------	-----	------

Threshold for symmetric handover detection

passfitpoints	no	int	20	none
----------------------	----	-----	----	------

Number of STSP packets to use to fit a pass

postpassdelay	no	int	15	none
----------------------	----	-----	----	------

Number of STSP packets to ignore after a pass when determining the delta

searchjumps	no	bool	no	none
--------------------	----	------	----	------

Diagnostic: Search for H/O jumps. If set to yes, then can use another optional param, jumpsearchwindow.

jumpsearchwindow	no	int	30	none
-------------------------	----	-----	----	------

Diagnostic: When searchjumps is yes, then Number of Station Points after H/O to search for jumps

usetimefilter	no	bool	no	none
----------------------	----	------	----	------

Use the Good Time Filter file.

timefilterfile	no	string	goodtime.fit	none
-----------------------	----	--------	--------------	------

When usetimefilter is yes, then File describing good time periods.

cachedir	no	string	/home/tcxgen/tm/rawtm/none	none
-----------------	----	--------	----------------------------	------

TM Cache directory

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.

The normal run will produce some warnings and even errors, which will not prevent the RTX file from being created.

Only those ERROR and WARNING in capital will be checked and will prevent the RTX of being placed in the final processed directory.

No data File (*error*)

Check the entered timespan and that the required Raw TM files are located in the cache dir directory.

High RMS (*warning*)

The fitting process is very poor. Beware if this is due to removal of too many points, fictitious passes because of interleaved TM or any Raw TM data rejection.



corrective action: High RMS No corrective action, but default parameters may be modified for improved fitting

OBT reset (*warning*)

Check the TM is ok, check the true OBT reset, otherwise force the RTX as good enough, or use the bad time intervals.

corrective action: an OBT reset is detected because the counter does not increase as expected. It may happen that a true OBT reset is present or any Raw TM data rejection process led to the apparent OBT reset.

6 Input Files

1. delays.fit, which contain the ground station delays, both the XMCS delays and the Actual (real) delays.
2. raw tm data files, as needed.
3. bti.fit, the bad time intervals, which is used for discarding ad hoc time intervals.
4. stations.sdid, which contains the stations data required by FDS routine SIGDEL (signal propagation delay).
5. orbita, which contains the orbital data required by FDS routine SIGDEL (signal propagation delay).
6. revno, which contains the revolutions start time.
7. goodtime.fit (optional), used with the corresponding task parameter about good time intervals.

7 Output Files

1. RTX_(rev)_(version).FIT, main output, containing the OBT-ERT fitted pairs.
2. RTX_(rev)_(version).log, log with the passes and fitting log info.
3. RTX_run_(rev)_(version).log, log with the stderr and stdout info.
4. RTX_(rev)_(version).stsp, with the stsp contents dump info.
5. RTX_(rev)_(version)_ORIGINAL.ps, with the original PREDUTC-OBTREAL vs. OBTREAL info.
6. RTX_(rev)_(version)_SYMFIX.ps, with the fitted, filtered and mark as bad PREDUTC-OBTREAL vs. OBTREAL info.

8 Algorithm

8.1 Phase I: packets are extracted

The master processing chain is as follows, - the raw file is opened. - The DU are extracted. - The Frames are extracted. Only VC 0 will be processed further. - The packets are extracted. Only STSP packets will be processed further. (see Phase II) (the overspilling is handled for the extraction)



The pairs FrameTime and Packet Time are read.

- The Frame Time is read from frame header. then is corrected taking into account OBDH delay, GS delay and signal delays.

- the packet time is calculated as per: $iOffset = FrameStatus.FirstHeader.iOffset$ gives the number of bytes not transmitted in the packet from the last frame $pktTime = frameTime + ((iOffset * 8) / 70000.0)$

- The Raw packet time is read from the APID 0 packet byte data, Both the S-Field, P-Field, Coarse and Fine - It may also extract the information from APID which contain time information in Data Field DFH (such as APID640 ACC OBT times) as raw fine and coarse information (only 1,1 TM is currently processed).

8.2 Phase II: STSP packets are processed

We have a dedicated STSP Processor.

When reading one per one packets, it checked the acceptPackett method. (i.e. check if it is an APID 0)

If so, the dedicated process packet for STSP method is called.

This method, - Process the Frame (unwraps the MFC from the frame, and calculate the VC0 time) - Process the packet (generate the STSP record)

The STSP Packet Processor get the current Frame Time and calculate the trigger time (packet time, corrected from XMCS OBDH delays, XMCS GS delays, Actual OBDH delays, Actual GS delays, and propagation delays). The STSP record is written. It will contain,

- Time Coarse
- Time Fine
- Time Total
- TimeVC0
- Current Frame Time
- Signal Delay
- Signal Delay2
- G/S
- Actual GS delay (SLE=0)
- OBDH delay
- Corrected Trigger Time (ERT, Corrected STSP Time)
- Corrected Trigger Time

This information is sent to the list of STSP packets, which will be used for the final fitting process (aka: RTX file creation)



8.3 Phase III: the fitting process

. The RTX is built from reading pairs of, - OBT (OBT Coarse and Fine) - ERT (corrected STSP time)

The RTX files are created, based in the STSP packet lists in memory. the OBT-UTC pairs are read from those memory lists, check GS passes, write the first version of RTX file containing OBT Coarse, OBT Fine, Days, CDS milliseconds, ERT, number of Pass, GSID, etc...

Finally, the STSP prior info is corrected (correct STSP process). So the RTX file is re-opened, and then a fitting process, spike removal, etc... are launched.

9 Comments

- Note that this task is purely SOC internal, not intended for general usage.

References